Validation of Independent Components using a Hypothesis Testing Approach

Corine de Koker

Thesis presented in the partial fulfilment
of the requirement for the degree of
MCom (Mathematical Statistics)
at the University of Stellenbosch

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF.

Supervisors : Dr. David Hofmeyr & Mr. Hans-Peter Bakker

Degree of confidentiality: A

November 2020

PLAGIARISM DECLARATION

1. Plagiarism is the use of ideas, material and other intellectual property of another's work and to present it as my own.
2. I agree that plagiarism is a punishable offence because it constitutes theft.
3. Accordingly, all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.
4. I also understand that direct translations are plagiarism.
5. I declare that the work contained in this thesis, except otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this thesis or another thesis.

C. De Koker	26 November 2020
Initials and surname	Date

ACKNOWLEDGEMENTS

The author would like to acknowledge Dr. David Hofmeyr, as well as Mr. Hans-Peter Bakker for their guidance in creating this thesis. Furthermore, the author would like to acknowledge the financial assistance of the National Research Foundation (NRF) towards this research. The author would also like to acknowledge the Department of Statistics and Actuarial Science for making available a research thesis template.

ABSTRACT

The main focus of this thesis is the validation of Independent Component Analysis (ICA), a popular technique used in signal processing. In a typical application, the purpose of ICA is to extract non-Gaussian signals representing the source signals from observed signals that are mixtures of the source signals in the case where the source signals are unavailable or unknown. This thesis only considers the FastICA implementation of ICA in the case where the number of source signals are equal to the number of mixture signals, and where any additive noise can be neglected. The FastICA algorithm extracts non-Gaussian signals through the maxmisation of negentropy. The more non-Gaussian the source signals, the more closely the signals extracted using FastICA represent the source signals. Amongst other things, this thesis demonstrates a novel approach using hypothesis testing with negentropy as a test statistic to determine the degree of non-Gaussianity of the source signals. The results from the hypothesis test mentioned previously were compared to the results from a second hypothesis test which uses a measure suggested by Himberg *et al.* (2004) that measures the compactness of the clusters of estimates of ICA components. The clustering visualisation methods proposed by Himberg *et al.* (2004) were also executed in this thesis and provided visual support for the results from the hypothesis tests. Both hypothesis tests were performed on three different datasets. The first dataset contained mixtures of only non-Gaussian signals. The second dataset contained mixtures of three non-Gaussian and three Gaussian signals, while the third dataset contained mixtures of only Gaussian signals. Both hypothesis tests rejected the null hypothesis that each of the source signals contained in the dataset are Gaussian when applied to the first dataset, which is in line with our expectations. The results from both hypothesis tests indicated the presence of three Gaussian and three non-Gaussian source signals in the second dataset. Regarding the third dataset, both hypothesis tests rejected about 5% of the signals extracted by the FastICA algorithm, which was as expected since a significance level of 5% was used. Therefore, our results provide evidence that hypothesis testing could potentially be used as an alternative method to indicate the degree of non-Gaussianity of mixtures of source signals.

Key words:

ICA; Hypothesis testing; non-Gaussianity

OPSOMMING

Die fokus van hierdie tesis is die validering van Onafhanklike Komponent Analise (OKA), ‘n gewilde tegniek in seinprossesering. Die doel van OKA is om nie-Gaussiese seine wat die oorspronklike seine verteenwoordig te beraam wanneer net mengsels van die oorspronklike seine beskikbaar is. Hierdie tesis oorweeg net die FastICA implementasie van OKA in die geval waar die aantal oorspronklike seine gelyk is aan die aantal mengsel seine, en waar additiewe ruis nagelaat kan word. FastICA beraam nie-Gaussiese seine deur die maksimalisering van negentropie. Hoe meer nie-Gaussies die oorspronklike seine, hoe nader verteenwoordig die beramings van die FastICA algoritme die oorspronklike seine. Onder andere het hierdie tesis ‘n nuwe benadering gedemonstreer deur gebruik te maak van hipotese toetsing met negentropie as ‘n toetsstatistiek om die graad van nie-Gaussianiteit van die oorspronklike seine te bepaal. Die resultate van die voorgenoemde hipotese toets is vergelyk met die resultate van ‘n tweede hipotese toets wat gebruik maak van ‘n mate voorgestel deur Himberg *et al.* (2004) wat die kompakheid van groeperings van beramings van OKA komponente meet. Die groeperings-visualiseringsmetodes voorgestel deur Himberg *et al.* (2004) was ook uitgevoer in hierdie tesis en verskaf visuele ondersteuning vir die resultate van die hipotese toetse. Beide hipotese toetse is uitgevoer op drie verskillende datastelle. Die eerste datastel is saamgestel uit vermengings van slegs nie-Gaussiese seine. Die tweede datastel het bestaan uit vermengings van drie nie-Gaussiese en drie Gaussiese seine, terwyl die derde datastel slegs uit vermengings van Gaussiese seine bestaan het. Beide hipotese toetse het die nulhipotese - dat elke sein in die datastel Gaussies is - verwerp vir al die seine toe die algoritme toegepas was op die eerste datastel, wat volgens ons verwagtings is. Die resultate van beide hipotese toetse het nagenoeg drie Gaussiese en drie nie-Gaussiese seine aangedui in die tweede datastel. Aangaande die derde datastel het beide hipotese toetse 5% van die seine verwerp. Dit stem ooreen met wat verwag is, aangesien ‘n vertrouevlak van 5% gebruik was. Die gevolgtrekking is dus dat hipotese toetsing die potensiaal het om gebruik te kan word as ‘n alternatiewe metode om die graad van nie-Gaussianiteit van oorspronklike seine te bepaal, wat kan voorspel hoe akkuraat die beraamde seine ooreenstem met die oorspronklike seine.

Sleutelwoorde:

Onafhanklike Komponent Analise, Hipotese toetsing, nie-Gaussianiteit.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND/OR ACRONYMS

AL	Average Linkage
BSS	Blind Source Separation
CCA	Curvilinear Component Analysis
CL	Complete Linkage
ICA	Independent Component Analysis
JADE	Joint Approximate Diagonalisation of Eigenmatrices
MDS	Multidimensional Scaling
PCA	Principal Component Analysis
pdf	probability density function
SL	Single Linkage
SVD	Singular Value Decomposition

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In signal processing, it is often desirable to extract source signals from their mixtures when only their mixtures are available. The probability distributions of source signals can range from being Gaussian to being non-Gaussian. If all the signals are Gaussian, Principal Component Analysis (PCA) is often used to decorrelate the signals. However, for many real world problems, the Gaussian assumption is often invalid. The term Blind Source Separation (BSS) refers to the problem of finding the original source signals, given only observed signals that are assumed to be linear mixtures of the original source signals (Westad and Kermit, 2003). In other words, both the source signals and the mixing process are unavailable or unknown. Independent Component Analysis (ICA) is a BSS technique that attempts to recover the original signals by estimating a transformation that maximises statistical independence between the sources under the assumption that the data does not follow a Gaussian distribution. The more non-Gaussian a source signal, the closer the signals extracted by ICA will represent the original source signal. Since ICA is often applied to datasets containing mixtures of source signals whose distributions are unknown, the degree of non-Gaussianity of the source signals is unknown. This means that the signals extracted using ICA are not necessarily accurate or reliable representations of the original source signals. It might therefore be valuable to be able to determine the degree of non-Gaussianity of the source signals when the source signals are unknown. This thesis explores using the principles of hypothesis testing to indicate the degree of non-Gaussianity of source signals. This thesis is limited to the basic noise free, instantaneous ICA method where the time dimension is ignored. The hypothesis testing is demonstrated using three different datasets. The first dataset only contains non-Gaussian signals, the second a combination of non-Gaussian and Gaussian signals, and the third only Gaussian signals. The results from the hypothesis tests are then compared to the source signals, which are known for the purposes of this thesis, as well as to the clustering methods proposed by Himberg *et al.* (2004), which provide a graphical way by which departure from Gaussianity of the estimated components can be inspected.

1.2 PROBLEM STATEMENT

The problem being explored in this thesis is to use the principles of hypothesis testing to determine the degree of non-Gaussianity of source signals in the case where the distribution of the source signals are unknown.

1.3 CLARIFICATION OF KEY CONCEPTS

The first key concept that is necessary to clarify is BSS. As mentioned before, BSS refers to the problem of finding source signals given only observed signals that are mixtures of the original source signals (Westad and Kermit, 2003).

The second key concept is ICA. Also mentioned before, ICA is a BSS technique that attempts to recover the original signals by estimating a transformation that provides statistical independence between the sources under the assumption that the data does not follow a Gaussian distribution.

The third key concept is hypothesis testing. The type of hypothesis test applied in this thesis can be described as follows. To conduct a hypothesis test several components are needed, namely a null hypothesis, alternative hypothesis, null distribution, test statistic and significance level. In our case, the null hypothesis is a speculation about some parameter from a specific population. The alternative hypothesis is what we can understand about this parameter if the null hypothesis is rejected. The test statistic is the estimate of this parameter using a test dataset. The null distribution is created by estimating the parameter multiple times using samples of data from the population under the null hypothesis. The significance level is the probability of rejecting when the null hypothesis is true. The significance level determines the size of the rejection region. If the test statistic falls within the rejection region of the null distribution the null hypothesis is rejected.

The fourth key concept is clustering. Clustering is the grouping of similar objects into subsets such that those within each subset are more closely related to one another than objects assigned to different subsets.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

Before the use of the principles of hypothesis testing can be explored to determine the non-Gaussianity of the extracted signals, it is necessary to understand the theory that the techniques that were applied are based on. This chapter provides the necessary theory behind the ICA algorithm that was applied in this thesis.

2.2 BLIND SOURCE SEPARATION

ICA is perhaps the most widely used method for performing BSS (Hyvärinen and Oja, 2000). Here, a "source" means an original signal and "blind" means that little is known about how these original signals have been mixed together to produce the signals that we observe. More specifically, we only know that the observed signals are linear mixtures of the original signals. Recall that the aim is to extract independent components representing the original source signals from the mixtures that we observe, hence the term "separation". These techniques originated as an attempt to solve variants of the classic cocktail party problem. A simple example of this can be demonstrated as follows. Two people are having a conversation in a room with two microphones. The problem involves extracting each person's contribution to the conversation from the mixture signals recorded by the microphones. This example arises in acoustics, in which BSS has been applied widely (e.g. Parra and Spence (2000), Lee *et al.* (1998), Murata *et al.* (2001*a*), Parra *et al.* (2000), Murata *et al.* (2001*b*)). However, BSS techniques have been applied across various other domains as well. Besides acoustics, other common applications include biomedical signal processing, telecommunications and finance (Hyvärinen *et al.*, 2001), as well as image processing (Ruckebusch, 2016). BSS techniques have also been applied successfully in video stabilisation (Qureshi *et al.*), genetics (Pearlson *et al.*, 2015), mining (Lin, 2010), and even chemistry (Ruckebusch, 2016). Therefore, BSS techniques can be used in any domain where an array of receivers picks up mixtures of a number of source signals (Bell and Sejnowski, 1995*a*). The term "signal" can extend to other types of data and is not necessarily restricted to acoustic data or data from telecommunications in the application of

ICA.

Some notational conventions that will be used are: scalars in lower case, matrices in upper case, and vectors in boldface lowercase. The ith component of a vector, say $\mathbf{x}$, is denoted x_i and the ijth component of a matrix, say X, is denoted x_{ij}. The expectation operator is E and transposition is indicated by superscript T. All the vectors are interpreted to be column vectors, which means that the transpose of $\mathbf{x}$, $\mathbf{x}^T$, is a row vector. The identity matrix is denoted I, where its dimension will be clear from the context.

Now, in order to model the BSS problem, let a source signal be represented by a stationary process $s_i, i = 1, ..., N$. Similarly, let a mixture signal be represented by a stationary process $x_i, i = 1, ..., N$. Now, as mentioned previously, the BSS problem often involves multiple source signals and mixture signals. Hence, let $\mathbf{s}_i^T = [s_{i1}, s_{i2}, ..., s_{ip}]$ denote the vector of p source signals at time i. Similarly, let $\mathbf{x}_i^T = [x_{i1}, x_{i2}, ..., x_{iq}]$ denote the vector containing the q mixture signals at time i. BSS is then considered an unsupervised learning problem that seeks to find the p unknown source signals $s_{ij}, j = 1, ..., p$, (with time points $i = 1, ..., N$), when only their mixtures $x_{ik}, k = 1, ..., q$, are available (Meinecke *et al.*, 2002).

Now consider $q \geq p$ mixture signals, $\mathbf{x}_i^T, i = 1, ..., N$. Then according to a linear mixture model, each of the $x_{ik}, k = 1, ..., q$, is assumed to be generated by

$$x_{ik} = \sum_{j=1}^{p} a_{kj} s_{ij}, \tag{2.1}$$

where $a_{kj}, k = 1, ..., q, j = 1, ..., p$, are the elements of the mixing matrix A. In many applications, it would be realistic to assume that the measurements contain some noise (Hyvärinen (1998*a*), Hyvarinen (1999)). This would justify the addition of a noise term in the model. However, for simplicity this thesis only considers the case where any additive noise can be neglected. This is because the noise-free model is deemed to be sufficient for many applications (Hyvärinen and Oja, 2000). For the moment, it is also simpler to assume that A is square ($p \times p$). However, this assumption can be relaxed.

BSS techniques are based on the properties of source signals and mixtures signals, namely independence and Gaussianity (Stone, 2004). Source signals are assumed to be independent and

non-Gaussian, while mixture signals are not independent and generally closer to Gaussian. Different BSS techniques use different properties of source signals to extract the source signals from the mixture signals. As mentioned before, the focus of this thesis is on ICA, which is a BSS technique that involves maximising independence and/or non-Gaussianity, depending on the ICA implementation. For example, ICA can be implemented using FastICA (Hyvärinen and Oja, 2000), Infomax (Bell and Sejnowski, 1995*a*) or Joint Approximate Diagonalization of Eigenmatrices (JADE) (Cardoso and Souloumiac, 1993). The different ICA implementations have their own advantages and disadvantages. For example, the FastICA and Infomax algorithms perform gradient searches, while JADE does not. This means that JADE avoids convergence problems that sometimes occur in FastICA and Infomax. However, JADE is more computationally demanding for high-dimensional datasets, but faster for large N.

2.3 THE ICA MODEL

For notational convenience, and to simplify the derivation of the ICA model, the time index is now dropped since the time structure of the signals is not taken into account for the purposes of this thesis. The notation $s_j, j = 1, ..., p$ can then be interpreted as a random variable that can take on any one of the N values in the jth source signal, but is still referred to as a source signal. Similarly, $x_j, j = 1, ..., p$ represents a random variable that can take on any one of the N values in the jth mixture signal, and is referred to as a mixture signal.

Before the ICA model can be introduced, it is necessary to clarify the following. As mentioned before, the original source signals are unknown. ICA attempts to estimate the original source signals. Therefore, let $s_j, j = 1, ..., p$, denote the random variables representing the true source signals and let $\tilde{s}_j, j = 1, ..., p$, denote the random variables representing the independent components which are estimators of the original source signals. The pdfs of each of the $s_j, j = 1, ..., p$, and $\tilde{s}_j, j = 1, ..., p$, are unknown but assumed to be non-Gaussian. Lastly, let $\hat{s}_j, j = 1, ..., p$, denote the extracted signals, which are estimates of the independent components. The true mixture signals will be represented by $x_j, j = 1, ..., p$.

In order to introduce the ICA model, we need to assume that the source signals are statistically independent and that the mixing matrix A is of full column rank. In this case, the BSS problem

reduces to identifying the mixing matrix A using only the mixture signals, while assuming statistical independence of the source signals, as well as linear independence of the columns of A (Meinecke *et al.*, 2002). Since the true mixing matrix A is unknown, we will use $\tilde{A}$ to denote the estimator of A in the ICA model. A realisation of $\tilde{A}$ will then be denoted by $\hat{A}$.

The rigorous ICA definition can be given using a statistical 'latent variables' model (Comon (1994); Jutten and Herault (1991)). The term 'latent variables' refers to the fact that the variables in the model are not directly observed but are rather inferred. Consider the case where p linear mixtures $x_1, ..., x_p$ of p source signals are observed. Then the ICA model can be given as

$$x_j = \tilde{a}_{j1}\tilde{s}_1 + \tilde{a}_{j2}\tilde{s}_2 + ... + \tilde{a}_{jp}\tilde{s}_p, \tag{2.2}$$

for $j = 1, ..., p$ (Hyvärinen and Oja, 2000). Without loss of generality, it is assumed that the mixture variables, as well as the source have a mean of zero. In the case where this is not true, the observed variables can be centered by subtracting the sample mean, which makes the model zero-mean (Hyvärinen and Oja, 2000).

For the sake of convenience, the ICA model can also be expressed in vector-matrix notation. The above mixing model can then be written as

$$\mathbf{x} = \tilde{A}\tilde{\mathbf{s}}. \tag{2.3}$$

The main idea behind the ICA model is that if the matrix A can be estimated by $\tilde{A}$, and if its inverse, say $\tilde{W}$, can be computed, then the independent components can be obtained by

$$\tilde{\mathbf{s}} = \tilde{W}\mathbf{x}. \tag{2.4}$$

Note that the ICA problem is undetermined: since only the mixture signals $x_j, j = 1, ..., p$, are known, it is possible for a scalar factor to be exchanged between each independent component $\tilde{s}_j, j = 1, ..., p$, and the corresponding column of $\tilde{W}$ without the product being changed. For this reason, we cannot determine the variances of the independent components (Hyvärinen and Oja, 2000). As a consequence, the magnitudes of the independent components can be fixed by assuming

each has unit variance: $E\{\tilde{s}_j^2\} = 1, j = 1, ..., p$. Then the matrix $\tilde{W}$ will be adapted in the ICA solution methods to take this restriction into account. Note that this still leaves the ambiguity of the sign. Fortunately, it is insignificant in most applications (Hyvärinen and Oja, 2000).

Also note that the ordering of the independent components and the corresponding columns of $\tilde{W}$ has no meaning. This means that the independent components can only be recovered up to a permutation, scales and signs. In other words, only an unordered set of one-dimensional independent component subspaces can be identified (Meinecke *et al.*, 2002).

2.4 STATISTICAL INDEPENDENCE

The starting point for ICA is to assume that the source signals $s_j, j = 1, ..., p$, are statistically independent (Hyvärinen and Oja, 2000). This is because ICA essentially estimates the unmixing matrix, and therefore the independent components, by maximising the independence between the components in the mixtures of the source signals.

In order to define independence, consider two scalar-valued variables x_1 and x_2. In basic terms, these two variables are said to be statistically independent if none of them carries any information about the other (Westad and Kermit, 2003).

More formally, let x_1 and x_2 have a joint probability density of $f(x_1, x_2)$ and marginal probability densities $f_1(x_1)$ and $f_2(x_2)$, where

$$f_1(x_1) = \int f(x_1, x_2) dx_2, \tag{2.5}$$

and similarly for x_2. Then, x_1 and x_2 are independent if their joint probability density $f(x_1, x_2)$ is factorial, i.e.

$$f(x_1, x_2) = f_1(x_1) f_2(x_2). \tag{2.6}$$

This definition can be extended for any number, n, of random variables (Hyvärinen and Oja, 2000). In that case, the joint density is the product of the n marginal distributions.

It is worth noting that independence implies uncorrelatedness (Hyvärinen and Oja, 2000), which causes many ICA methods to constrain the estimation procedure so that it always gives uncorrelated estimates of the independent components. This simplifies the problem by reducing the number of

free parameters (Hyvärinen and Oja, 2000).

2.5 MEASURE OF INDEPENDENCE

In order for ICA to maximise the independence between the components in the mixture signals, a measure for independence needs to be defined. Unfortunately, this task turns out to be more challenging than one would expect. This is because independence itself cannot be measured easily. However, quantities related to independence can be measured, which prove to be useful.

One of these quantities is mutual information. Mutual information is a measure of the amount of information that one random variable contains about another random variable (Cover and Thomas, 1991). It is a natural measure of the dependence between random variables (Hyvärinen and Oja, 2000). The general idea behind using mutual information as a measure of independence is that variables that carry little information about each other would tend to be more independent than variables that carry a lot of information about each other. A more rigorous discussion follows.

In order to calculate the mutual information between two variables, a quantity called relative entropy is required. Before relative entropy can be described, it is necessary to define entropy. Entropy is a measure of the amount of information required on average to describe a random variable (Cover and Thomas, 1991). In statistics, this is interpreted as a measure of the uncertainty of a random variable. The more uncertain or unpredictable a random variable, the larger its entropy.

Mathematically, entropy can be expressed as follows. Let x be a continuous random variable with a probability density function f whose support is a set $\mathcal{X}$. Then the entropy of x is defined by (Cover and Thomas, 1991)

$$H(x) = -\int_{\mathcal{X}} f(x) \log\{f(x)\} dx. \tag{2.7}$$

This definition can be extended to a pair of random variables. The joint entropy $H(x, y)$ of a pair of continuous random variables (x, y) with a joint distribution $f(x, y)$ is defined as (Cover and Thomas, 1991)

$$H(x, y) = -\int_{\mathcal{X},\mathcal{Y}} f(x, y) \log\{f(x, y)\} dx dy. \tag{2.8}$$

Relative entropy is a measure of the distance between two distributions (Cover and Thomas, 1991).

The relative entropy between two probability density functions $f(x)$ and $g(x)$ is defined as

$$D(f||g) = \int_{\mathcal{X}} f(x) \log\{\frac{f(x)}{g(x)}\}dx = E_f \log\{\frac{f(x)}{g(x)}\}. \tag{2.9}$$

From Eq. 2.9 it can be seen that relative entropy is an expected logarithm of the likelihood ratio. Thus, relative entropy is always non-negative and is zero if and only if $f = g$. Relative entropy is also known as the Kullback-Leibler divergence.

Finally, mutual information can be defined. Consider two continuous random variables x and y with support sets $\mathcal{X}$ and $\mathcal{Y}$, a joint probability density function $f(x, y)$ and marginal probability density functions $f(x)$ and $f(y)$. The mutual information between x and y, $I(x; y)$, is the relative entropy between the joint density $f(x, y)$ and the product of the marginal densities $f(x)f(y)$, i.e. (Cover and Thomas, 1991),

$$I(x; y) = \int_{\mathcal{X},\mathcal{Y}} f(x, y) \log\{\frac{f(x, y)}{f(x)f(y)}\}dxdy = D(f(x, y)||f(x)f(y)) = E_{f(x,y)} \log\{\frac{f(x, y)}{f(x)p(y)}\}. \tag{2.10}$$

This definition can be extended to more than two variables. The mutual information $I(.)$ between m (scalar) random variables, $y_i, i = 1, ..., m$ is given as follows (Hyvärinen and Oja, 2000):

$$I(y_1, y_2, ..., y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{y}). \tag{2.11}$$

Mutual information is a very natural measure for independence (Hyvärinen and Oja, 2000). This can be seen as follows. Recall that two variables x_1 and x_2 with a joint probability density of $f(x_1, x_2)$ and marginal probability densities $f_1(x_1)$ and $f_2(x_2)$ are independent if their joint probability density $f(x_1, x_2)$ is factorial (see Eq. 2.6). Also recall that this definition can be extended for any number n of random variables. From Eq. 2.10 it can be seen that this means that the mutual information between independent variables is zero.

The Infomax ICA algorithm (Bell and Sejnowski, 1995a) uses mutual information and entropy to obtain the independent components. Bell and Sejnowski (1995a) and Nadal and Parga (1994) derived this algorithm from a neural network point of view (Hyvärinen and Oja, 2000). The Infomax algorithm performs stochastic gradient ascent in the mutual information between inputs and outputs

of a network (Bell and Sejnowski, 1995*b*). By maximising the mutual information between inputs and outputs, the network 'factorises' the input into independent components. Maximum likelihood estimation is another popular approach for estimating the ICA model (Hyvärinen and Oja, 2000). It is closely related to the Infomax principle since it can be shown that both the Infomax and ML approaches to ICA actually lead to exactly the same equation to be optimised. This is because both methods depend on the assumption that the pdfs of the independent components is the same as the pdfs of the required source signals (Stone, 2004). Unfortunately, this assumption is quite unrealistic since the pdfs of the source signals are not known in general. Despite this, it has been found that ICA works because if the model pdfs are approximations to the source pdfs it yields extracted signals that approximates the original source signals (Cardoso (2000); Amari (1998)).

The FastICA approach, which is applied in this thesis, is based on the concept of non-Gaussianity, which will be discussed next.

2.6 NON-GAUSSIANITY

In the previous section, it was described how mutual information as a measure of independence can be used as one way to obtain the independent components. Another way to obtain the independent components is to use non-Gaussianity. According to Bell and Sejnowski (1995*a*), minimising the mutual information between the mixture signals and the extracted signals has the same effect as minimising the entropy of the extracted signals. Maximising the non-Gaussianity of the extracted signals is motivated by the fact that Gaussian variables have the largest entropy of all variables with equal variance (Hyvärinen and Oja, 2000). This can also be justified using the Central Limit Theorem. Recall that the distributions of independent components are assumed to be non-Gaussian. According to the Central Limit Theorem, the distribution of a sum of independent random variables tends toward a Gaussian distribution under certain conditions (Hyvärinen and Oja, 2000). Therefore, the sum of two independent variables will usually have a distribution that is closer to a Gaussian distribution than either one of the original random variables.

This can be applied to ICA as follows. Let $\mathbf{x}$ be a vector containing the p random variables representing the mixture signals that are distributed according to the ICA model. For simplicity, assume for now that all the independent components have identical distributions. In order to

estimate one of the independent components $\tilde{s}_1$, consider a linear combination of the $x_j, j = 1, ..., p$, denoted by $\tilde{s}_1 = \tilde{\mathbf{w}}_1^T \mathbf{x}$, where $\tilde{\mathbf{w}}_1$ is a vector to be determined. Let $\mathbf{z} = \tilde{A}^T \tilde{\mathbf{w}}_1$. Then $\tilde{s}_1 = \tilde{\mathbf{w}}_1^T \mathbf{x} = \tilde{\mathbf{w}}_1^T \tilde{A}\tilde{\mathbf{s}} = \mathbf{z}^T \tilde{\mathbf{s}}$. Therefore, $\tilde{s}_1$ is a linear combination of $\tilde{s}_j, j = 1, ..., p$, with weights given by $z_j, j = 1, ..., p$. Since a sum of two or more random variables is more Gaussian than the individual variables, $\mathbf{z}^T \tilde{\mathbf{s}}$ is more Gaussian than any of the $\tilde{s}_j$ and becomes less Gaussian when it equals one of the $\tilde{s}_j$. In this case, only one of the elements of $\mathbf{z}$ is non-zero (Hyvärinen and Oja, 2000). In other words, when the non-Gaussianity of $\mathbf{z}^T \tilde{\mathbf{s}}$ is maximised, it leads to the observation of the independent component that was intended to be estimated. Now since $\mathbf{z}^T \tilde{\mathbf{s}} = \tilde{\mathbf{w}}_1^T \mathbf{x}$, maximising the non-Gaussianity of $\mathbf{z}^T \tilde{\mathbf{s}}$ is the same as maximising the non-Gaussianity of $\tilde{\mathbf{w}}_1^T \mathbf{x}$. Thus, if the vector $\tilde{\mathbf{w}}_1$ can be found such as to maximise the non-Gaussianity of $\tilde{\mathbf{w}}_1^T \mathbf{x}$, it would lead to the independent component that was intended to be estimated.

2.6.1 Measures of Non-Gaussianity

In order to use non-Gaussianity to obtain the independent components, measures of non-Gaussianity are required. Two popular measures of non-Gaussianity will be discussed, namely excess kurtosis and negentropy (Hyvärinen and Oja, 2000).

2.6.1.1 Kurtosis

Excess kurtosis is the classical measure of non-Gaussianity (Hyvärinen and Oja, 2000). Excess kurtosis is the normalised fourth moment defined mathematically as follows. Let y be a random variable with zero mean and unit variance. Then the excess kurtosis of y is defined as

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 = E\{y^4\} - 3. \tag{2.12}$$

What makes excess kurtosis useful as a measure of non-Gaussianity is that it is zero for Gaussian random variables. This is because the fourth moment of a Gaussian variable y, $E\{y^4\}$, is equal to $3(E\{y^2\})^2$ (Hyvärinen and Oja, 2000). Thus

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 = 3(E\{y^2\})^2 - 3(E\{y^2\})^2 = 0. \tag{2.13}$$

Kurtosis can be positive or negative. Positive kurtosis is obtained from variables that are super-Gaussian. These variables have pdfs that are heavy at the tails, but smaller at intermediate values. Sound waves are an example of super-Gaussian data. Negative kurtosis is obtained from sub-Gaussian variables, for example the uniform distribution. Therefore, the absolute value or the square of kurtosis is typically used to measure non-Gaussianity.

The advantage of using kurtosis as a measure of non-Gaussianity is that it is simple to calculate both computationally and theoretically (Hyvärinen and Oja, 2000). The disadvantage of using kurtosis is that it can be very sensitive to outliers (Huber, 1985). This means that kurtosis is not a robust measure of non-Gaussianity, which is concerning (Hyvärinen and Oja, 2000). Therefore, it is worth investigating more reliable measures of non-Gaussianity for the ICA application since it is an optimisation problem which relies quite heavily on the robustness of the measure of non-Gaussianity.

2.6.1.2 Negentropy

Negentropy is another important measure of non-Gaussianity. Recall that entropy was discussed earlier and that Gaussian variables have the largest entropy among all random variables of equal variance (Cover and Thomas (2012), Papoulis and Pillai (2002)). This means that the Gaussian distribution is the "most random" or least structured distribution compared to all the other distributions. Therefore, entropy can also be used as a measure of non-Gaussianity.

Entropy is small for distributions that are clearly clustered or concentrated on certain values, and large for variables with Gaussian distributions (Hyvärinen and Oja, 2000). However, it would be desirable to find a transformation of entropy such that it is zero for Gaussian variables and non-zero for non-Gaussian variables, similar to kurtosis. Hence, negentropy, a modified version of entropy, can be used instead.

Negentropy J is defined as follows:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \tag{2.14}$$

where $\mathbf{y}_{gauss}$ is a Gaussian random vector of the same covariance matrix as $\mathbf{y}$ (Hyvärinen and Oja, 2000). Negentropy is always non-negative and is zero if and only if $\mathbf{y}$ has a Gaussian distribution.

Negentropy is an attractive measure of non-Gaussianity because of its desirable statistical properties (Hyvärinen and Oja, 2000). Unfortunately, it is very challenging to compute. This is because an estimate of the pdf would be required, which is generally not available. As a result, simpler approximations of negentropy have been developed.

2.6.1.3 Approximations of Negentropy

The classical method of approximating negentropy is using higher-order moments (Hyvärinen and Oja, 2000). As before, let y be a random variable with zero mean and unit variance. Then one example of the approximation to negentropy can be given as follows (Jones and Sibson, 1987):

$$J(y) \approx \frac{1}{12}E\{y^3\}^2 + \frac{1}{48}kurt(y)^2. \tag{2.15}$$

Unfortunately, approximations such as these suffer from the same issue of non-robustness encountered with kurtosis (Hyvärinen and Oja, 2000). In order to address this problem, new approximations were developed by Hyvärinen (1998*b*). In general, the following approximation is obtained

$$J(y) \approx \sum_{k=1}^{K} c_k[E\{G_k(y)\} - E\{G_k(v)\}]^2, \tag{2.16}$$

where $c_k, k = 1, ..., K$, are some positive constants, v is a standardised Gaussian variable, and the functions $G_k, k = 1, ..., K$, are some non-quadratic functions (Hyvärinen and Oja, 2000). This approximation is consistent in the sense that it is always non-negative, and in the case where y has a Gaussian distribution, it is equal to zero.

If only one non-quadratic function G is used, the approximation simplifies to

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2, \tag{2.17}$$

for practically any non-quadratic function G (Hyvärinen and Oja, 2000). G can be chosen to obtain improved approximations of negentropy. It has been found that if G is chosen such that it does not grow too fast, more robust estimators are obtained. The following choices of G have proved very

useful (Hyvärinen and Oja, 2000):

$$G_1(u) = \frac{1}{a} \log \cosh au, \tag{2.18}$$

$$G_2(u) = \exp(-u^2/2), \tag{2.19}$$

where $1 \leq a \leq 2$ is some suitable constant.

These approximations of negentropy are a good compromise between the properties of the two classical non-Gaussianity measures given by kurtosis and negentropy. They are also conceptually simple, fast to compute, and have appealing statistical properties, especially robustness (Hyvärinen and Oja, 2000). Therefore, these *contrast functions* will be used in the ICA algorithm applied in this thesis to obtain the independent components. The $\log \cosh$ function in Eq. 2.18 was chosen to be used as G in this thesis.

2.7 PREPROCESSING FOR ICA

Now that a measure for non-Gaussianity has been selected to optimise in the ICA algorithm, we only need to discuss the preprocessing of the data before the algorithm can be described. Preprocessing techniques make the problem of ICA estimation simpler and better conditioned (Hyvärinen and Oja, 2000).

2.7.1 Centering

Centering is the most basic and necessary preprocessing step (Hyvärinen and Oja, 2000). If $\mathbf{x}$ is the vector containing the mixed signals on which one wishes to apply ICA, $\mathbf{x}$ is centered by subtracting its mean vector $\mathbf{m} = E\{\mathbf{x}\}$, making it a zero-mean variable. This implies that $\tilde{\mathbf{s}}$, the vector containing the independent components, is a zero-mean variable as well.

Note that centering only simplifies the ICA algorithm; the algorithm can still be applied without centering the data (Hyvärinen and Oja, 2000). After $\tilde{A}$ is estimated with the centered data, the estimation is completed by adding the mean vector of $\tilde{\mathbf{s}}$, $\tilde{A}^{-1}\mathbf{m}$, back to the centered estimates of $\tilde{\mathbf{s}}$, where $\mathbf{m} = E\{\mathbf{x}\}$. This can be seen as follows: let $\mathbf{x}_c$ be the centered data vector of $\mathbf{x}$, i.e. $\mathbf{x}_c = \mathbf{x} - \mathbf{m}$. The centered estimates of $\tilde{\mathbf{s}}$, $\tilde{\mathbf{s}}_c$, are then obtained by $\tilde{\mathbf{s}}_c = \tilde{A}^{-1}\mathbf{x}_c$. Thus,

$\tilde{\mathbf{s}}_c = \tilde{A}^{-1}\mathbf{x}_c = \tilde{A}^{-1}(\mathbf{x} - \mathbf{m}) = \tilde{A}^{-1}\mathbf{x} - \tilde{A}^{-1}\mathbf{m} = \tilde{\mathbf{s}} - \tilde{A}^{-1}\mathbf{m}$. Therefore, $\tilde{\mathbf{s}} = \tilde{\mathbf{s}}_c + \tilde{A}^{-1}\mathbf{m}$, with $\tilde{A}^{-1}\mathbf{m}$ being the mean vector of $\tilde{\mathbf{s}}$.

In the rest of the thesis, it is assumed that the data have been centered. Thus, $\mathbf{x}$ will imply $\mathbf{x}_c$.

2.7.2 Whitening

Whitening refers to the linear transformation of $\mathbf{x}$ such that its components are uncorrelated with unit variance (Hyvärinen and Oja, 2000). The reason for performing whitening is that it reduces the number of parameters to be estimated. This can be illustrated as follows. A popular method to perform whitening is to use the eigenvalue decomposition of the covariance matrix $E\{\mathbf{x}\mathbf{x}^T\} = V\Lambda V^T$, where V is the orthogonal matrix of eigenvectors of $E\{\mathbf{x}\mathbf{x}^T\}$ and Λ is the diagonal matrix of its eigenvalues, $\Lambda = diag(\lambda_1, ..., \lambda_p)$. Whitening is performed as follows: let $\mathbf{x}_w$ denote $\mathbf{x}$ that is whitened, then $\mathbf{x}_w = V\Lambda^{-1/2}V^T\mathbf{x}$, where the matrix $\Lambda^{-1/2}$ is computed by a simple component-wise operation as $\Lambda^{-1/2} = diag(\lambda_1^{-1/2}, ..., \lambda_p^{-1/2})$. Now, $E\{\mathbf{x}_w\mathbf{x}_w^T\} = I$. The usefulness of whitening arises from the fact that the new $\tilde{A}_w$ is orthogonal. This can be seen as follows: $E\{\mathbf{x}_w\mathbf{x}_w^T\} = \tilde{A}_w E\{\tilde{\mathbf{s}}\tilde{\mathbf{s}}^T\}\tilde{A}_w^T = \tilde{A}_w\tilde{A}_w^T = I$. Now, only the new, orthogonal $\tilde{A}_w$, with $p(p-1)/2$ degrees of freedom, has to be estimated, as opposed to the p^2 parameters of the original $\tilde{A}$.

It can also be useful to perform dimension reduction together with whitening. This can be done by discarding the eigenvalues λ_j of $E\{\mathbf{x}\mathbf{x}^T\}$ that are too small, similar to performing principal component analysis. This often reduces noise and prevents overfitting, which has been observed in ICA (Hyvärinen *et al.*, 1999).

In the rest of the thesis, it is assumed that the data have been preprocessed by centering and whitening.

2.7.3 Further preprocessing

It is worth mentioning that the performance of ICA can be improved by performing application-dependent preprocessing (Hyvärinen and Oja, 2000). However, this will not be discussed further in this thesis since no application-dependent preprocessing was performed on the data used in this thesis. This allows for the application of ICA in this thesis to be fully data driven and universally applicable.

2.8 THE FASTICA ALGORITHM

Now that all the necessary mathematics have been explained, the algorithm for performing ICA can be discussed. As mentioned earlier, there exist more than one algorithm that can perform ICA. However, for the application in this thesis, only the FastICA algorithm was used. Therefore, only this algorithm will be described in detail.

First, ICA for one unit will be discussed. Take for example the vector $\mathbf{x}$ containing a random variable representing each of the p mixture signals, with a weight vector $\tilde{\mathbf{w}}_1$. The FastICA algorithm finds a unit vector $\tilde{\mathbf{w}}_1$ such that the projection $\tilde{\mathbf{w}}_1^T\mathbf{x}$ maximises non-Gaussianity (Hyvärinen and Oja, 2000). Recall that non-Gaussianity will be measured by the approximation of negentropy given in Eq. 2.17, with contrast function given in Eq. 2.19.

Let g denote the derivative of the non-quadratic function G used in Eq. 2.17. For example, the derivatives of Eq. 2.18 and Eq. 2.19 are:

$$g_1(u) = tanh(au), \tag{2.20}$$

$$g_2(u) = u\exp(-u^2/2), \tag{2.21}$$

where $1 \leq a \leq 2$ is some suitable constant, often taken as $a = 1$.

It has been found that the maxima of the approximation of the negentropy of $\tilde{\mathbf{w}}_1^T\mathbf{x}$ are obtained at certain optima of $E\{G(\tilde{\mathbf{w}}_1^T\mathbf{x})\}$ (Hyvärinen and Oja, 2000). According to the Kuhn-Tucker conditions (Luenberger, 1997), the optima of $E\{G(\tilde{\mathbf{w}}_1^T\mathbf{x})\}$ under the constraint $E\{(\tilde{\mathbf{w}}_1^T\mathbf{x})^2\} = ||\tilde{\mathbf{w}}_1||^2 = 1$ are obtained at points where

$$E\{\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - \beta\tilde{\mathbf{w}}_1 = 0, \tag{2.22}$$

where $g = G'$ and $\beta = E\{\tilde{\mathbf{w}}_1^T\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\}$.

Eq. 2.22 can be solved using Newton's method. Recall that Newton's method can be given as follows. Let $F : R^p \to R^p$ be a differentiable function. We seek a solution to $F(\mathbf{x}) = \mathbf{0}$, starting from an initial point $\mathbf{x} = \mathbf{x}_1$. At the nth step, given $\mathbf{x}_n$, the next approximation can be computed

by

$$\mathbf{x}_{n+1} = \mathbf{x}_n - [JF(\mathbf{x}_n)]^{-1} F(\mathbf{x}_n), \tag{2.23}$$

where JF is the Jacobian matrix of F. Eq. 2.23 is then iterated until convergence.

Newton's method can be applied to Eq. 2.22 as follows. Let $F(\tilde{\mathbf{w}}_1) = E\{\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - \beta\tilde{\mathbf{w}}_1$. Then the Jacobian matrix of $F(\tilde{\mathbf{w}}_1)$, $JF(\tilde{\mathbf{w}}_1)$, can be obtained as $JF(\tilde{\mathbf{w}}_1) = E\{\mathbf{x}\mathbf{x}^T g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - \beta I$. Recall that the Jacobian matrix $JF(\tilde{\mathbf{w}}_1)$ contains the partial derivatives of $F(\tilde{\mathbf{w}}_1)$. Since the inversion of this matrix is required in the iteration, it is useful to simplify the matrix using an approximation. A reasonable approximation can be given as $E\{\mathbf{x}\mathbf{x}^T g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\} \approx E\{\mathbf{x}\mathbf{x}^T\}E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\} = E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\}I$ because of the spherical nature of the data (Hyvärinen and Oja, 2000). This results in the Jacobian matrix becoming diagonal, which can easily be inverted. Therefore, the following Newton iteration is obtained (Hyvärinen and Oja, 2000):

$$\tilde{\mathbf{w}}_1^+ = \tilde{\mathbf{w}}_1 - [E\{\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - \beta\tilde{\mathbf{w}}_1]/[E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - \beta]. \tag{2.24}$$

This algorithm can be simplified further by multiplying both sides of the equation by $\beta - E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\}$. After algebraic simplification, this gives the FastICA iteration (Hyvärinen and Oja, 2000)

$$\tilde{\mathbf{w}}_1^+ = E\{\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\}\tilde{\mathbf{w}}_1. \tag{2.25}$$

The basic form of the FastICA algorithm can then be given as follows (Hyvärinen and Oja, 2000):

1. Choose an initial weight vector $\tilde{\mathbf{w}}_1$.
2. Let $\tilde{\mathbf{w}}_1^+ = E\{\mathbf{x}g(\tilde{\mathbf{w}}_1^T\mathbf{x})\} - E\{g'(\tilde{\mathbf{w}}_1^T\mathbf{x})\}\tilde{\mathbf{w}}_1$
3. Let $\tilde{\mathbf{w}}_1 = \tilde{\mathbf{w}}_1^+ / ||\tilde{\mathbf{w}}_1^+||$
4. If not converged, go back to 2.

Note that convergence means that the old and new values of $\tilde{\mathbf{w}}_1$ point in the same direction. In other words, their dot-product is close to 1 (Hyvärinen and Oja, 2000).

In the practical implementation of the algorithm, the expectations are replaced by their estimates. The natural estimates are the corresponding sample means. Let $\hat{\mathbf{w}}_1$ denote $\tilde{\mathbf{w}}_1$ in Eq. 2.25 when the expectations are replaced by their estimates. Let $\hat{\mathbf{w}}_1^*$ denote the $\hat{\mathbf{w}}_1$ at which the algorithm

converges. The estimate of the ith observation contained in one independent component can then be given as

$$\hat{s}_{i1} = \hat{\mathbf{w}}_1^{*T}\mathbf{x}_i, \tag{2.26}$$

where $\mathbf{x}_i^T$ is the ith row of the $N \times p$ matrix X containing the N observations of each of the p observed mixture signals.

The one-unit algorithm above only estimates one of the independent components. In order to estimate p independent components, the one-unit FastICA algorithm needs to be adjusted such as to estimate $\tilde{\mathbf{w}}_j^*, j = 1, ..., p$ (Hyvärinen and Oja, 2000). This can be done as follows.

In order to prevent different vectors from converging to the same maxima, the outputs $\tilde{\mathbf{w}}_1^T\mathbf{x}, ..., \tilde{\mathbf{w}}_p^T\mathbf{x}$ have to be decorrelated after every iteration. There are two common methods for achieving this (Hyvärinen and Oja, 2000). Note that since $\mathbf{x}$ is assumed to be whitened, decorrelation in this case is equivalent to orthogonalisation.

The first method being described to decorrelate the outputs is a deflation scheme. This means that the independent components are estimated one by one. Suppose r independent components have been estimated and the one-unit algorithm above is run for $\tilde{\mathbf{w}}_{r+1}$. In order to ensure that the final estimate of $\tilde{\mathbf{w}}_{r+1}$, $\hat{\mathbf{w}}_{r+1}^*$, is orthogonalised, after every iteration step in the algorithm, the "projections" $\hat{\mathbf{w}}_{r+1}^{*T}\hat{\mathbf{w}}_j{}^*\hat{\mathbf{w}}_j^*, j = 1, ..., r$ of the previously estimated r vectors are subtracted from $\hat{\mathbf{w}}_{r+1}^*$, and then $\hat{\mathbf{w}}_{r+1}^*$ is renormalised. This can be summarised in the following steps:

1. Let $\hat{\mathbf{w}}_{r+1}^* = \hat{\mathbf{w}}_{r+1}^* - \sum_{j=1}^{r} \hat{\mathbf{w}}_{r+1}^{*T}\hat{\mathbf{w}}_j^*\hat{\mathbf{w}}_j^*$
2. Let $\hat{\mathbf{w}}_{r+1}^* = \hat{\mathbf{w}}_{r+1}^*/||\hat{\mathbf{w}}_{r+1}^*||$

In order to estimate p independent components, the one-unit algorithm is run for $\tilde{\mathbf{w}}_j, j = 1, ..., p$ to obtain $\hat{\mathbf{w}}_j^*, j = 1, ..., p$, where the deflation scheme is applied after every $\hat{\mathbf{w}}_j^*, j = 1, ..., p$ was obtained from the algorithm. The ith observation of the jth independent component can then be given as

$$\hat{s}_{ij} = \hat{\mathbf{w}}_j^{*T}\mathbf{x}_i, \tag{2.27}$$

where $\mathbf{x}_i^T$ is the ith row of the $N \times p$ matrix X containing the N observations of each of the p observed mixture signals. This method was applied in this thesis.

In certain applications, it is more desirable for the independent components to be estimated si-

multaneously. This method can be given by the following iterative algorithm (Hyvärinen and Oja, 2000):

1. Choose an initial matrix $\tilde{W} = (\tilde{\mathbf{w}}_1, ..., \tilde{\mathbf{w}}_p)^T$
2. Let $\tilde{W}^+ = E\{Xg(\tilde{W}X)\} - E\{g'(\tilde{W}X)\}\tilde{W}$
3. Let $\tilde{W} = (\tilde{W}^+\tilde{W}^{+T})^{-1/2}\tilde{W}^+$, where the inverse square root $(\tilde{W}^+\tilde{W}^{+T})^{-1/2}$ is obtained from the eigenvalue decomposition of $\tilde{W}^+\tilde{W}^{+T} = U\Lambda U^T$ as $(\tilde{W}^+\tilde{W}^{+T})^{-1/2} = U\Lambda^{-1/2}U^T$.
4. If not converged, go back to 2.

As before, the ith observation of the jth independent component can then be given as

$$\hat{s}_{ij} = \hat{\mathbf{w}}_j^{*T}\mathbf{x}_i, \tag{2.28}$$

where $\mathbf{x}_i$ is the ith row of the $N \times p$ matrix X containing the N observations of each of the p observed mixture signals and $\hat{\mathbf{w}}_j^*$ is the jth row of $\hat{W}^*$, which is the value of $\tilde{W}$ for which the algorithm above converged.

The FastICA algorithm can also be expressed such that the connection to the Infomax or maximum likelihood algorithm introduced in Amarai *et al.* (1996), Bell and Sejnowski (1995*a*), Cardoso and Laheld (1996), and Cichocki and Unbehauen (1996) is apparent. This can be done as follows. Eq. 2.25 can be written in matrix form as (Hyvärinen and Oja, 2000)

$$\tilde{W}^+ = \tilde{W} + diag(\alpha_j)[diag(\beta_j) + E\{g(\tilde{\mathbf{s}})\tilde{\mathbf{s}}^T\}]\tilde{W}, \tag{2.29}$$

where $\tilde{\mathbf{s}} = \tilde{W}\mathbf{x}$, $\beta_j = -E\{\tilde{s}_j g(\tilde{s}_j)\}, j = 1, ..., p$, and $\alpha_j = -\frac{1}{\beta_j + E\{g'(\tilde{s}_j)\}}, j = 1, ..., p$. This form is similar to the stochastic gradient method for maximising likelihood in Eq. 2.30

$$\tilde{W}^+ = \tilde{W} + \mu[I + g(\tilde{\mathbf{s}})\tilde{\mathbf{s}}^T]\tilde{W}, \tag{2.30}$$

where μ is the learning rate, not necessarily constant in time, and g is a function of the pdfs of the independent components: $g = f_j'/f_j, j = 1, ..., p$, where f_j is the pdf of the jth independent component (Hyvärinen and Oja, 2000).

2.8.1 FastICA applied to Gaussian data

It is worth mentioning that FastICA cannot accurately estimate the Gaussian signals if more than one Gaussian signal is present in the data. This is because the Gaussian joint densities are symmetrical which causes any orthogonal transformation to have exactly the same distribution and there is no information on the directions of the columns of the mixing mixing matrix. The data is centered and whitened before FastICA is applied, therefore the estimates are realised as the random initialisations of the algorithm that are iterated until the threshold of the maximum number of iterations in the algorithm is reached. This causes the estimates to be random which results in the estimates for the different Gaussian signals being indistinguishable. FastICA can accurately estimate the non-Gaussian signals in the presence of Gaussian signals, but the estimates of the Gaussian signals would be indistinguishable.

CHAPTER 3

VALIDATION OF THE ICA ALGORITHM

3.1 INTRODUCTION

Recall that the purpose of applying ICA is to extract non-Gaussian signals from mixtures of source signals. The FastICA algorithm is based on the assumptions that the source signals are non-Gaussian and statistically independent, as mentioned in the previous chapter. However, it is possible for mixtures of source signals to contain Gaussian, as well as non-Gaussian signals. The more non-Gaussian the source signals, the closer the estimates from the FastICA algorithm would represent the source signals. Therefore, in the case where the source signals are unknown, it might be desirable to be able to determine the non-Gaussianity of the source signals.

3.2 VALIDATION OF ICA IN THE LITERATURE

According to Westad and Kermit (2003), only a very limited number of studies on model validation have been reported in the ICA literature, and that those that have been reported are poorly described. Westad and Kermit (2003) proposed using the variance of the columns of the estimated unmixing matrices to determine the number of non-Gaussian independent components present in the data. The columns of an estimated unmixing matrix can also be referred to as ICA loadings. Cross-validation was used to indicate the variance of the ICA loadings as follows. Consider the $N \times p$ data matrix X. Suppose that X is segmented into K matrices of size $\frac{N}{K} \times p$ and that ICA is performed on X, as well as K more times, leaving out one of the segments from the full data matrix each time. Let $\hat{\mathbf{w}}_j$ denote the estimate of the jth component using the full data matrix and let $\hat{\mathbf{w}}_{j(-k)}$ denote the estimate of the jth component using the data matrix without the kth segment, $k = 1, ..., K$. Then the variance of the jth loading can be given as follows (Efron, 1982)

$$s^2(\hat{\mathbf{w}}_j) = \frac{K-1}{K} \sum_{k=1}^{K} (\hat{\mathbf{w}}_j - \hat{\mathbf{w}}_{j(-k)})^2, \tag{3.1}$$

where $s^2(\hat{\mathbf{w}}_j)$ is the variance of the jth loading. A small variance would indicate a more non-Gaussian independent component present in the data, while a large variance would indicate a more

Gaussian component in the data. This will be elaborated on in a later section.

Meinecke *et al.* (2002) used resampling to estimate the stability of the estimated independent components. Similar to above, more stable estimates of the independent components would indicate that the underlying independent components are more non-Gaussian. Again, this will be elaborated on in a later section. Himberg *et al.* (2004) incorporated the works of Westad and Kermit (2003) and Meinecke *et al.* (2002) to look at the clustering of the ICA loadings as an indication of the non-Gaussianity of the independent components. Tight clusters would indicate independent components that are more non-Gaussian, while more spread out clusters would indicate components that are more Gaussian. In this thesis we explore the application of the principles of hypothesis testing as an indication of the non-Gaussianity of the underlying independent components. The clustering methods suggested by Himberg *et al.* (2004) were also applied in order to compare the results from the hypothesis tests.

3.3 HYPOTHESIS TESTING

Hypothesis testing can provide a method of identifying non-Gaussian source signals with statistical significance. In this thesis, we will explore hypothesis testing using negentropy as a test statistic, as well as I_q, which is a measure of the compactness of the clusters of the ICA loadings. Recall that the signals extracted by the FastICA algorithm are realisations of independent components in the data, which are estimators of the true source signals. The hypothesis test using negentropy will be performed on the extracted signals, while the hypothesis test using I_q will be performed on the ICA loadings.

First, hypothesis testing using negentropy as a test statistic will be discussed. Recall that negentropy is zero for Gaussian variables. This fact can be used in hypothesis testing to determine whether an signal extracted by the FastICA algorithm is non-Gaussian. This can be done as follows. First, let us consider the case where only one signal is extracted by the algorithm. The null hypothesis is that the negentropy of this extracted signal is equal to zero. Mathematically this can be expressed as $H_0 : J(\tilde{s}) = 0$, where $J(.)$ denotes the negentropy approximation and $\tilde{s}$ denotes the signal extracted by the FastICA algorithm. The alternative hypothesis is that the negentropy of the extracted signal is greater than zero, i.e. $H_1 : J(\tilde{s}) > 0$. Therefore, if the null hypothesis is

rejected, we can conclude that the extracted signal is non-Gaussian.

The test statistic used in the hypothesis test is the negentropy approximation $J(\hat{\mathbf{s}}) \propto [E\{G(\hat{\mathbf{s}})\} - E\{G(\mathbf{v})\}]^2$, where $G(\mathbf{v}) = -\log\cosh(-\mathbf{v}^2/2)$, $\mathbf{v}$ is a vector of standardised Gaussian variables and $\hat{\mathbf{s}}$ is the extracted signal. Recall from Section 2.6.1.3 that an alternative G can be used in the negentropy approximation which could cause the results from this thesis to differ. Now, in order to calculate the rejection region, we need to determine the values of $J(\hat{\mathbf{s}})$ for which the null hypothesis will be rejected. This can be done as follows. The FastICA algorithm can be applied to a dataset that does not contain any non-Gaussian signals to extract one signal. The negentropy of the signal extracted by the algorithm is then recorded. This process is repeated to generate a distribution of the negentropy measurements recorded, which will result in a sample from the *null distribution* for the hypothesis test. For the sake of convenience, for the rest of this thesis these samples of the true null distributions are referred to as null distributions. After the measurements have been sorted from smallest to largest, the rejection region is the top $100(\alpha)\%$ of the null distribution. That is, if the test statistic falls within the top $100(\alpha)\%$ of the recorded measurements, the null hypothesis will be rejected.

More often, the FastICA algorithm is applied to extract multiple signals. In this case, the hypothesis test above can be applied individually to each signal extracted by the FastICA algorithm. The only adjustment that has to be made to the hypothesis test above in the case of multiple signals is to consider the ordering of the negentropy of the extracted signals. In order to form the null distributions, the recorded negentropy measurements at each repetition can be sorted from smallest to largest. The test statistics can also then be sorted from smallest to largest and the hypothesis tests can be performed using the corresponding test statistics and null distributions.

The FastICA algorithm together with the hypothesis test described above can be applied multiple times on the same dataset to form a distribution of the number of extracted signals for which the null hypothesis was rejected. This should give a clearer indication of the number of non-Gaussian independent components present in the data, and thus the number of non-Gaussian source signals.

3.4 VARIABILITY OF EXTRACTED SIGNALS

Another way to assess the non-Gaussianity of the underlying independent components (and thus the source signals) is to look at the variability of the extracted signals. According to Himberg *et al.* (2004), two major factors that could affect the variability of the extracted signals are the algorithmic and statistical reliability of the algorithm. The algorithmic reliability of the algorithm refers to the effect that the stochastic nature of the algorithm has on the results. Recall from the previous chapter that the FastICA algorithm estimates the unmixing matrix by starting with a random initial matrix and then iterating the algorithm until convergence. This means that the results may be somewhat different in different runs of the algorithm. On the other hand, the finite sample sizes induce statistical errors in the estimation, which is what the statistical reliability would indicate (Meinecke *et al.*, 2002).

First, let us consider the algorithmic reliability of the FastICA algorithm. Himberg *et al.* (2004) found that the algorithmic reliability of the FastICA algorithm can be improved by running the algorithm multiple times, say B times, on the same dataset, starting at different initial points. This means that say p signals will be extracted B times. Meinecke *et al.* (2002) and Himberg *et al.* (2004) found that the statistical reliability of the FastICA algorithm can be assessed using bootstrapping. Bootstrapping is a well-known computational method for computing the statistical reliability in the case where a simple mathematical formula cannot be found (Tibshirani and Efron, 1993). Bootstrapping is a resampling method. This means that the data sample is randomly changed by simulating the sampling process, and the algorithm is then run many times with the bootstrapped samples that are somewhat different from each other. The reliability of the original estimate can then be analysed by looking at the spread of the obtained estimates (Himberg *et al.*, 2004). This can be applied to the FastICA algorithm as follows. The statistical reliability of the FastICA algorithm can also be improved by applying the algorithm multiple times, say B times. However, instead of starting at different initial points, the algorithm is applied to different random samples of the original dataset every time. Himberg *et al.* (2004) suggest combining both of the above methods to ensure both algorithmic and statistical reliability of the algorithm. In other words, the algorithm is applied multiple times, say B times, to different random samples of the original dataset, starting at different initial points every time. This method was applied in this

thesis.

Now that the algorithmic and statistical reliability of the algorithm has been accounted for, the variability of the extracted signals can be investigated. Since the FastICA algorithm maximises non-Gaussianity, when the true signals are Gaussian, the variation in the extracted signals is higher. This is because after whitening, the Gaussian estimates are invariant to rotation, which means that the optimal projections will be roughly uniformly distributed on the unit sphere. Thus, the variability of the estimates could give an indication of the non-Gaussianity of the underlying independent components. One way of exploring the variability of the estimates is through the use of clustering. This will be discussed in the next section.

3.5 CLUSTERING

According to Friedman *et al.* (2001), clustering relates to grouping or segmenting a collection of objects into subsets or "clusters", such that those within each cluster are more closely related to one another than objects assigned to different clusters. This calls for some measure of relation between the objects, commonly referred to as a dissimilarity measure. For the purposes of this thesis, we would like to investigate the clustering of the ICA loadings. Assuming that an independent component would be associated with each cluster, tight clusters would indicate that the estimates in the clusters do not vary much and that their underlying independent components are more non-Gaussian. On the other hand, clusters that are more spread out would indicate that the underlying independent components are more Gaussian.

The dissimilarity measure applied in this thesis is based on the mutual correlation coefficients of the estimates of the unmixing matrix. This dissimilarity measure was suggested by Himberg *et al.* (2004). It can be calculated as follows. Suppose that the algorithm is run B times on the $N \times p$ data matrix X. The estimates of unmixing matrices $\hat{W}_b$ from each run $b = 1, 2, ..., B$ are collected into a single matrix $\hat{W} = [\hat{W}_1 \hat{W}_2 ... \hat{W}_B]$. If p independent components are estimated on each round, we get Bp estimates, and the size of $\hat{W}$ will be $p \times Bp$. Now, a natural measure of similarity between the estimates of the unmixing matrix is the absolute value of their mutual correlation coefficients $r_{ij}, i, j = 1, ..., Bp$ (Himberg *et al.*, 2004). The final similarity matrix then has the elements σ_{ij}

defined by (Himberg *et al.*, 2004)

$$\sigma_{ij} = |r_{ij}|. \tag{3.2}$$

The similarity matrix can then be transformed into a dissimilarity matrix with elements (Everitt *et al.*, 1993)

$$\delta_{ij} = 1 - \sigma_{ij}. \tag{3.3}$$

3.5.1 Clustering techniques

In order to cluster the ICA loadings, a clustering technique is required. For the purposes of this thesis, only agglomerative hierarchical clustering was applied. Agglomerative hierarchical clustering was applied by Himberg *et al.* (2004), motivated by the fact that it is a well-known method for a modest number of objects (Everitt *et al.* (1993); Gordon (1987)).

Hierarchical clustering is named after the hierarchical representation produced by the method. This hierarchical structure emerges as the clusters at each level of the hierarchy are created by merging clusters at the next lower level (Friedman *et al.*, 2001). This hierarchical representation is otherwise known as a *dendrogram*.

The algorithm can be given as follows (Friedman *et al.*, 2001):

1. Suppose we have n observations and a measure of all the $\binom{n}{2}$ pairwise dissimilarities. Treat each observation as its own cluster.
2. For $i = n, n-1, \ldots, 2$:
   (a) Examine all pairwise inter-cluster dissimilarities among the i clusters and identify the pair of clusters that are least dissimilar. Fuse these two clusters. The dissimilarity between these two clusters indicates the height in the dendrogram at which the fusion should be placed.
   (b) Compute the new pairwise inter-cluster dissimilarities among the $i-1$ remaining clusters.

If one or both of the clusters contain multiple observations, the concept of dissimilarity between a pair of observations needs to be extended to a pair of groups of observations. This extension is achieved by developing the notion of linkage (Friedman *et al.*, 2001). There exist three common types of linkage. Complete Linkage (CL) computes all pairwise dissimilarities between the observa-

tions in cluster A and the observations in cluster B, and records the largest of these dissimilarities. Single Linkage (SL) computes all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and records the smallest of these dissimilarities. Average Linkage (AL) computes all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and records the average of these dissimilarities. Himberg *et al.* (2004) found that AL is the best to use with the FastICA algorithm. This is because SL is in general reported to be more sensitive to noise than AL and CL (Everitt *et al.*, 1993) and the experiments Himberg *et al.* (2004) conducted revealed that when the number of clusters is smaller than the data dimension, CL starts to join clusters inconsistently.

3.5.2 Clustering quality measures

After the clustering has been performed, it would be valuable to have a measure that could quantify the compactness of each of the clusters. Himberg *et al.* (2004) suggested using I_q as such a measure. This measure was also applied in this thesis. I_q is a conservative cluster quality index that reflects the compactness and isolation of a cluster (Himberg *et al.*, 2004). It is computed as the difference between the average intracluster similarities and average intercluster similarities. Intracluster similarities are the similarities between the points in a cluster, while intercluster similarities are the similarities between points in a cluster and points not in that cluster. Mathematically, I_q can be derived as follows.

As before, suppose that the FastICA algorithm is run B times on the $N \times p$ data matrix X. The estimates of unmixing matrices $\hat{W}_b$ from each run $b = 1, 2, ..., B$ are collected into a single matrix $\hat{W} = [\hat{W}_1 \hat{W}_2 ... \hat{W}_B]$. The similarity and dissimilarity measures are given in Eq. 3.2 and Eq. 3.3 respectively. For the purposes of this thesis, the number of clusters was taken as the number of mixture signals p.

Now, I_q can be given as follows. Let C denote the set of Bp indices of the columns of $\hat{W}$, let C_k denote the set of these indices that belong to the kth cluster and let $|C_k|$ denote the size of the kth cluster. Then the cluster quality index I_q is given as follows (Himberg *et al.*, 2004):

$$I_q(C_k) = \frac{1}{|C_k|^2} \sum_{i,j \in C_k} \sigma_{ij} - \frac{1}{|C_k||C_{-k}|} \sum_{i \in C_k} \sum_{j \in C_{-k}} \sigma_{ij}, \tag{3.4}$$

where C_{-k} is the set of indices that do not belong to the kth cluster and σ_{ij} is defined as in Eq. 3.2. Eventually, $I_q(C_k)$ is equal to one for an ideal cluster when Eq. 3.2 is used to compute the similarities σ_{ij}, and decreases when C_k becomes less compact and isolated.

3.5.2.1 Hypothesis testing

Similar to the hypothesis test performed using negentropy, hypothesis testing can also be performed using I_q to give an indication of the non-Gaussianity of the underlying independent components in the data. This can be done as follows.

For simplicity, first consider the case in which we would like the FastICA algorithm to only extract one signal. Suppose that we perform the FastICA algorithm on a set of p N-dimensional mixture signals B times to obtain a $p \times B$ $\hat{W}$ matrix, resampling from the set of mixtures and changing the initial unmixing matrix every time. Then we can calculate the I_q of this cluster of B estimates using the similarity measure in Eq. 3.2. This will be our test statistic for the hypothesis test. Now, in order to generate the first observation of the null distribution, we sample from a standard multivariate normal distribution to generate p N-dimensional Gaussian signals. Then we apply the FastICA algorithm B times to obtain a $p \times B$ $\hat{W}$ matrix and calculate the I_q of the cluster, similar to above. We repeat this process M times to form the null distribution. After the I_q measurements have been sorted from smallest to largest, the rejection region is the top $100(\alpha)\%$ of the null distribution. Again, if the test statistic falls within the top $100(\alpha)\%$ of the recorded measurements, the null hypothesis will be rejected. In this case, the null hypothesis is that the independent component present in the data is Gaussian. If we reject the null hypothesis, we can conclude that the independent components is non-Gaussian.

As before, the FastICA algorithm is more often applied to extract multiple signals. In this case, we perform the FastICA algorithm on the set of p N-dimensional mixture signals B times, extracting p signals every time, resulting in a $p \times Bp$ $\hat{W}$ matrix. These Bp column estimates would form at most p clusters in p-dimensional space. We can then calculate the I_q for each cluster using the similarity measure in Eq. 3.2 and perform the hypothesis test above separately on each cluster. Similar to the previous hypothesis test, an adjustment is made regarding the ordering of the I_q measurements. In order to form the null distribution corresponding to each cluster, the recorded I_q measurements at each repetition when simulating the null distribution can be sorted from smallest to largest.

The test statistics can also then be sorted from smallest to largest and the hypothesis tests can be performed using the corresponding test statistics and null distributions. If the hypothesis test is rejected for a cluster, it means that that cluster represents a non-Gaussian independent component present in the data.

Again, the hypothesis test described above can be applied multiple times on the same dataset to form a distribution of the number of extracted signals for which the null hypothesis was rejected. This could give a clearer indication of the non-Gaussianity of the underlying independent components, and thus the source signals.

Himberg *et al.* (2004) mention another quantitative index that could be applied to measure the partitioning of the clusters, namely I_R. It is often used as a quantitative index for suggesting the number of clusters that best fits the data (Himberg *et al.*, 2004). This measure could also be explored as an alternative to I_q in the hypothesis test. However, only I_q was applied for the purposes of this thesis.

3.5.3 Clustering visualisation

Visualisation is a popular method to inspect the distribution of points in clusters. As mentioned before, the result of the hierarchical clustering is typically presented as a dendrogram. A dendrogram demonstrates the clustering of points by successively joining the points in the clusters when moving upwards in the dendrogram. The vertical axis gives the dissimilarity for which the clusters are merged. The heights at which the clusters form in the dendrogram give an indication of the compactness of the clusters. If the points are joined at a higher level in the vertical axis, the dissimilarity is larger, which means that the points in the clusters are further away from each other and the clusters are less compact than if the points were joined at a lower level in the dendrogram. For the purposes of this thesis, we would expect clusters of estimates of Gaussian source signals to be less compact and therefore join higher up in the dendrogram compared to clusters of estimates of non-Gaussian signals.

Other types of visualisation also exist. Multidimensional Scaling (MDS) is a projection method that provides a useful visual one-, two- or three-dimensional representation of higher dimensional data. This is done by approximating the original dissimilarities between observations by Euclidean

distances in one, two or three dimensions.

According to Torgerson (1952), multidimensional scaling involves three basic steps. First, a scale of comparative distances between all pairs of points is obtained. In the ICA application in this thesis, the dissimilarity measure in Eq. 3.3 will be used. The second step involves estimating an additive constant and using this estimate to convert the comparative distances into absolute distances. Since the dissimilarity measure in Eq. 3.3 is based on the absolute value of the mutual correlation coefficients, which lie between zero and one, it already has a true zero point, which makes it an absolute distance. Lastly, the dimensionality of the space necessary to account for the absolute distances is determined, and the projections of the points on the axes of this space are obtained (Torgerson, 1952).

There exist several multidimensional scaling techniques. Himberg *et al.* (2004) compared three methods, namely Classical Scaling, Metric Least Squares Scaling and Curvilinear Component Analysis (CCA), for the dissimilarity measure in Eq. 3.3 using a trustworthiness index proposed by Venna and Kaski (2001). They found that CCA outperformed the other two methods according to this trustworthiness index in their experiments.

For the purposes of this thesis, CCA was used to visualise the clusters of the estimates of the unmixing matrix. The projection can be further controlled by modifying the definition of dissimilarity in Eq. 3.3 suitably, for example, as (Himberg *et al.*, 2004)

$$d_{ij}^{*} = \sqrt{1 - \sigma_{ij}}. \tag{3.5}$$

The definition of dissimilarity in Eq. 3.5 was used in the visualisation of the estimates of the unmixing matrix in this thesis, because it spreads the distribution of the distances so that differences in size among the most compact clusters can be seen better (Himberg *et al.*, 2004).

3.5.3.1 Curvilinear Component Analysis

We are seeking m-dimensional representations of the p-dimensional estimates of the columns of the $p \times Bp$ $\hat{W}$ matrix, where $m \leq p$. In Curvilinear Component Analysis (CCA) the extracted signals are seen as inputs to a neural network, while the m-dimensional projections $\hat{\mathbf{z}}_j, j = 1, ..., Bp$ are seen as the outputs. Consider the dissimilarities $\{\delta_{rs}\}$ as given in Eq. 3.5 and the Euclidean

distance d_{rs} between the outputs $\hat{\mathbf{z}}_r$ and $\hat{\mathbf{z}}_s$, $r, s = 1, ..., Bp$. The goal is to force d_{rs} to match δ_{rs} for each possible pair (r, s) (Demartines and Hérault, 1997). Since a perfect matching is not possible at all scales when manifold "unfolding" is needed to reduce the dimension from p to m, a weighting function $F(d_{rs}, \lambda)$ is introduced, yielding the quadratic cost function (Demartines and Hérault, 1997)

$$S = \frac{1}{2} \sum_r \sum_{s \neq r} (\delta_{rs} - d_{rs})^2 F(d_{rs}, \lambda). \tag{3.6}$$

Generally, $F(d_{rs}, \lambda)$ is chosen as a bounded and monotonically decreasing function, in order to favour local topology conservation (Demartines and Hérault, 1997). Local topology conservation refers to the preservation of the relationship between the estimates when they are projected. Decreasing exponential, sigmoid or Lorentz functions are all suitable choices.

The minimsation of the cost function with respect to the $\hat{\mathbf{z}}_j$'s is done by gradient descent. First, we rewrite Eq. 3.6 as a sum of partial costs (Demartines and Hérault, 1997)

$$S = \frac{1}{2} \sum_r \sum_{s \neq r} S_{rs}, \tag{3.7}$$

where $S_{rs} = (\delta_{rs} - d_{rs})^2 F(d_{rs}, \lambda)$.

Now, we need to find the derivative of Eq. 3.7 with respect to $\hat{\mathbf{z}}_s, s = 1, ..., p$,

$$\frac{\partial S_{rs}}{\partial \hat{\mathbf{z}}_s} = \frac{\partial S_{rs}}{\partial d_{rs}} \frac{\partial d_{rs}}{\partial \hat{\mathbf{z}}_s} \tag{3.8}$$

and since

$$d_{rs} = ((\hat{\mathbf{z}}_r - \hat{\mathbf{z}}_s)^T (\hat{\mathbf{z}}_r - \hat{\mathbf{z}}_s))^{\frac{1}{2}} \tag{3.9}$$

$$= (\hat{\mathbf{z}}_r^T \hat{\mathbf{z}}_r + \hat{\mathbf{z}}_s^T \hat{\mathbf{z}}_s - 2\hat{\mathbf{z}}_r^T \hat{\mathbf{z}}_s)^{\frac{1}{2}}, \tag{3.10}$$

$$\frac{\partial d_{rs}}{\partial \hat{\mathbf{z}}_s} = \frac{1}{2} (\hat{\mathbf{z}}_r^T \hat{\mathbf{z}}_r + \hat{\mathbf{z}}_s^T \hat{\mathbf{z}}_s - 2\hat{\mathbf{z}}_r^T \hat{\mathbf{z}}_s)^{-\frac{1}{2}} (2\hat{\mathbf{z}}_s - 2\hat{\mathbf{z}}_r) \tag{3.11}$$

$$= \frac{\hat{\mathbf{z}}_s - \hat{\mathbf{z}}_r}{d_{rs}}. \tag{3.12}$$

Now,

$$\frac{\partial S_{rs}}{\partial d_{rs}} = -2(\delta_{rs} - d_{rs})F(d_{rs}, \lambda) + (\delta_{rs} - d_{rs})^2 \frac{\partial}{\partial d_{rs}} F(d_{rs}, \lambda) \tag{3.13}$$

$$= -2(\delta_{rs} - d_{rs})F(d_{rs}, \lambda) + 0 \tag{3.14}$$

if we consider a quantised version of $F(d_{rs}, \lambda)$, which means that $\frac{\partial}{\partial d_{rs}} F(d_{rs}, \lambda) = 0$.

Eq. 3.8 then becomes

$$\frac{\partial S_{rs}}{\partial \hat{\mathbf{z}}_s} = -2(\delta_{rs} - d_{rs})F(d_{rs}, \lambda)\frac{\partial d_{rs}}{\partial \hat{\mathbf{z}}_s} \tag{3.15}$$

$$= -2(\delta_{rs} - d_{rs})F(d_{rs}, \lambda)\frac{\hat{\mathbf{z}}_s - \hat{\mathbf{z}}_r}{d_{rs}}. \tag{3.16}$$

In order to find the configuration $\hat{Z} : Bp \times m$ by minimising Eq. 3.6 using stochastic gradient descent, for $\hat{\mathbf{z}}_s, s = 1, ..., Bp$,

1. Initialise $\hat{\mathbf{z}}_s$ by sampling from a uniform distribution
2. Let

$$\hat{\mathbf{z}}_s^+ = \hat{\mathbf{z}}_s - \alpha(t)\frac{\partial S_{rs}}{\partial \hat{\mathbf{z}}_s}, \tag{3.17}$$

where $\alpha(t)$ is the learning rate that decreases with time, for example $\alpha(t) = \alpha_0/(1 + t)$ (Demartines and Hérault, 1997), i.e.

$$\hat{\mathbf{z}}_s^+ = \hat{\mathbf{z}}_s - \alpha(t)(-2(\delta_{rs} - d_{rs})F(d_{rs}, \lambda)\frac{\hat{\mathbf{z}}_s - \hat{\mathbf{z}}_r}{d_{rs}}) \tag{3.18}$$

$$= \hat{\mathbf{z}}_s + 2\alpha(t)(\delta_{rs} - d_{rs})F(d_{rs}, \lambda)\frac{\hat{\mathbf{z}}_s - \hat{\mathbf{z}}_r}{d_{rs}}. \tag{3.19}$$

In order to visualise the clusters more clearly on the MDS projection, a *convex hull* can be used to bound the estimates belonging to the same cluster (Gordon, 1987). Loosely speaking, a convex hull in this case is an envelope that contains the points in a cluster. A projection with a smaller convex hull should represent a more compact cluster such that an ideal cluster contracts into a single point (Himberg *et al.*, 2004). The similarities σ_{ij} can also be visualised rather explicitly by connecting

points with lines whose thickness/colour represent the similarities between then (Himberg *et al.*, 2004). This was also applied in this thesis, with more detail provided in the next two chapters.

The MDS plot with the convex hull applied in this thesis can be illustrated as follows. Suppose that the clustering is performed on the $p \times BP$ $\hat{W}$ matrix obtained after FastICA was applied B times to a $N \times p$ data matrix X. Using CCA, the two-dimensional coordinates of the square-root of the dissimilarities can be calculated using the correlation of the $\hat{W}$ matrix. These points can then be plotted and a convex hull can be used to bound the estimates belonging to the same cluster. In order to make the MDS plot more interpretable, some similarities can be shown by additional lines connecting the points. These lines can be drawn in different colours between dots representing estimates whose correlation (in absolute value) exceeds a certain threshold. For the purposes of this thesis, compact clusters would represent estimates of non-Gaussian source signals, while clusters containing points that are far from one another would represent estimates of Gaussian source signals.

CHAPTER 4

RESULTS

4.1 INTRODUCTION

This chapter is structured as follows. First, the three datasets that the hypothesis tests and visualisations were performed on are described. Next, estimates of the independent components representing the source signals in the first two datasets are visualised. Following this, the results from the hypothesis tests and visualisations for each of the three datasets are presented.

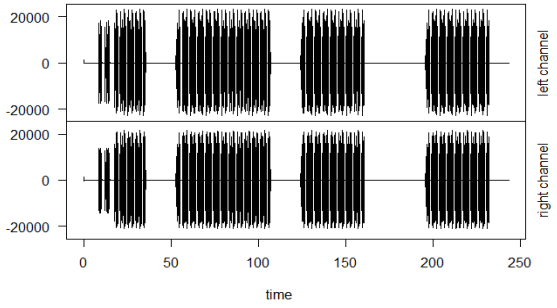
4.2 DATASETS

The validation methods were tested on three different types of datasets. The first dataset only contained non-Gaussian signals, the second dataset contained Gaussian as well as non-Gaussian signals, and the last dataset only contained Gaussian signals. These datasets will be referred to as the Non-Gaussian dataset, the Combination dataset and the Gaussian dataset, respectively.

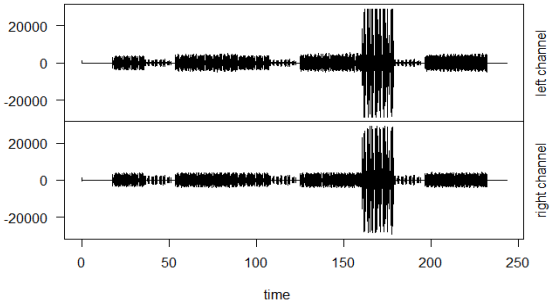
4.2.1 Non-Gaussian dataset

The non-Gaussian dataset was generated as follows. Seven non-Gaussian and mutually independent source signals were obtained as stems from a song. The song is called *Make it out Alive* and was created during a Producing and Beatmaking Masterclass by Timbaland. The song is made up of seven stems containing the drums, bass, voice, chords, special effects and chops. The time series representations of these signals are given in Figure 4.1. The marginal distributions of the same signals are given in Figure 4.2. From Figure 4.2 it is clear that the source signals are non-Gaussian, because the observations are concentrated at zero.

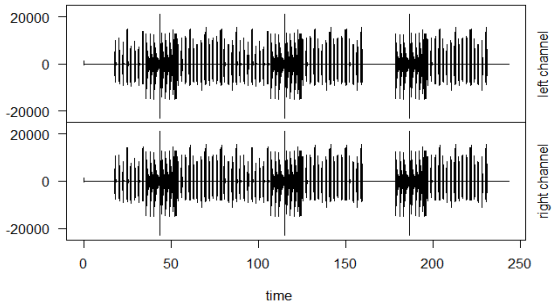
In order to perform the FastICA algorithm to extract estimates of these source signals, the signals have to be mixed together to obtain seven mixture signals. In order to do this, seven different combinations of six of the seven signals were mixed together. This was done by using Ableton Live, music producing software. A screenshot of the program containing an example of the six of the stems being mixed together is given in Figure 4.3. The six mixture signals were then exported as .wav files and imported into R. A link containing the six mixture signals and the R code necessary



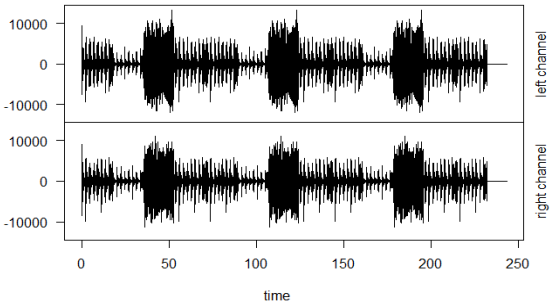
(a) Drums A



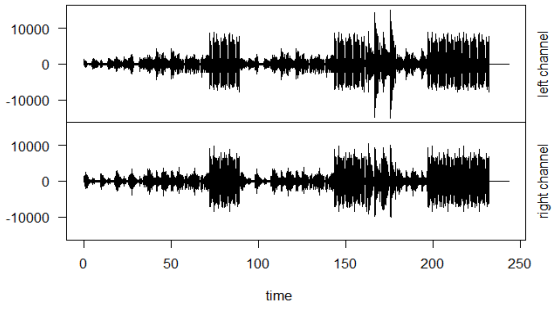
(b) Drums B



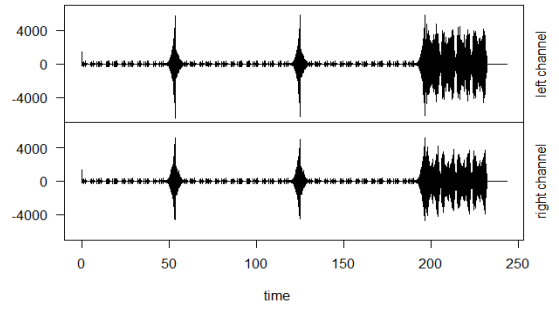
(c) Bass



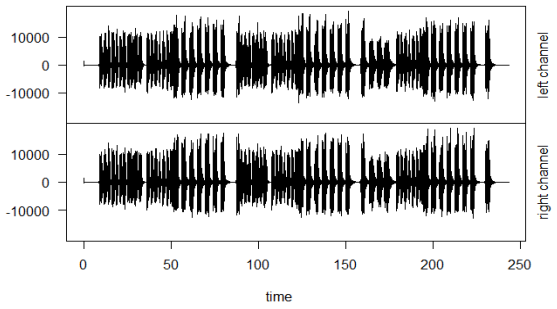
(d) Chords



(e) Chops

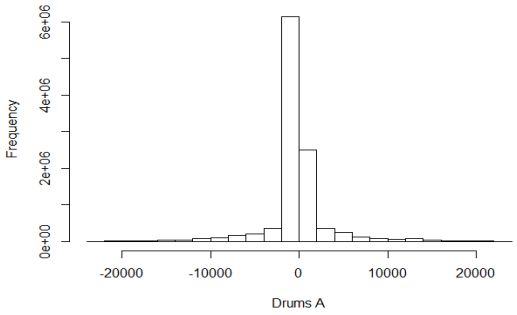


(f) SFX

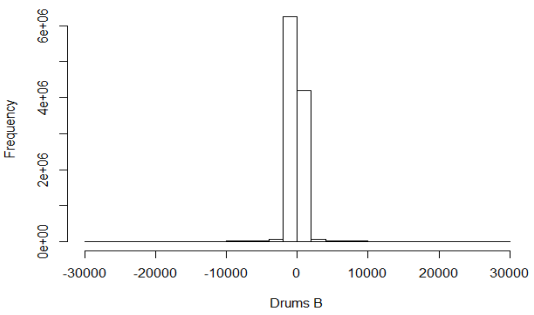


(g) Voice

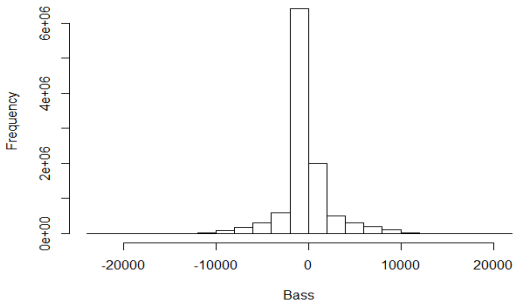
Figure 4.1: Time series representations of the seven non-Gaussian source signals



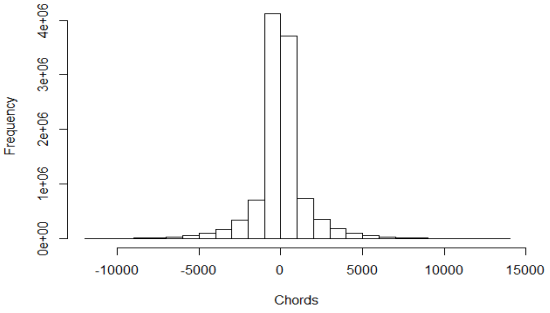
(a) Drums A



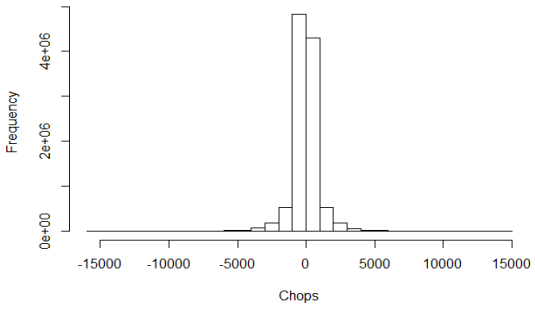
(b) Drums B



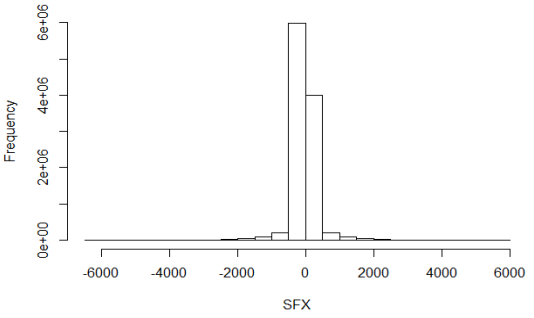
(c) Bass



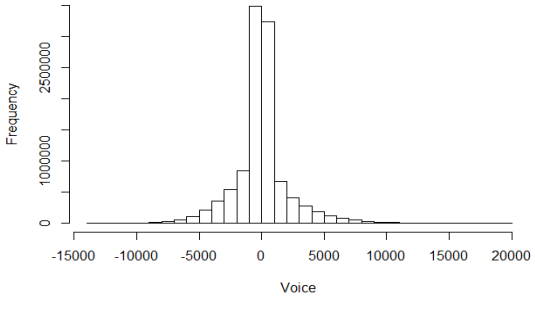
(d) Chords



(e) Chops



(f) SFX



(g) Voice

Figure 4.2: Marginal distributions of the seven non-Gaussian source signals

to reproduce the results is given in the appendix. This set of non-Gaussian mixture signals will be referred to as the non-Gaussian dataset. The hypothesis test were performed on a random sample of 2205 observations (equivalent to 50 ms) of each of the signals in this dataset.

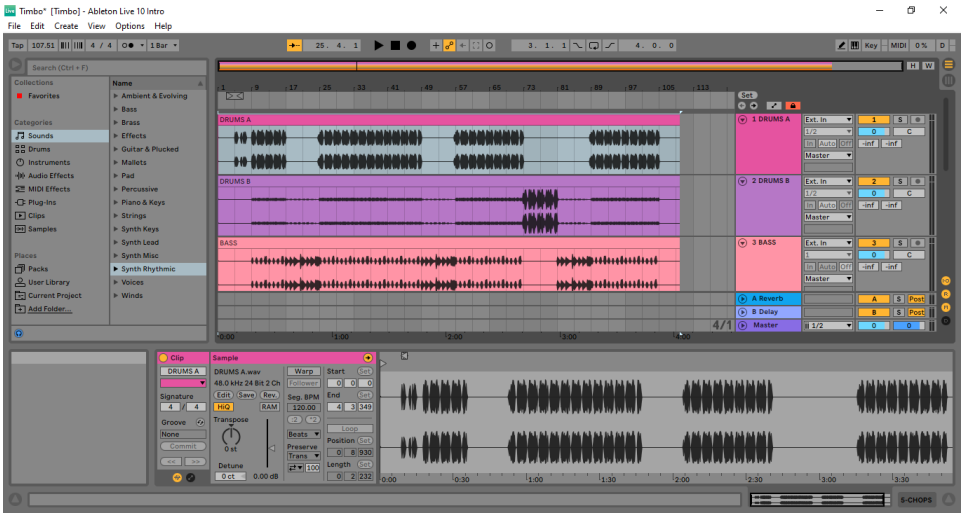


Figure 4.3: Screenshot using Ableton Live to mix six of the seven stems to produce a mixture signal

4.2.2 Combination dataset

The dataset containing both Gaussian and non-Gaussian source signals was generated as follows. First, the source signals were generated. Three non-Gaussian and three Gaussian source signals were generated using a software program called Audacity. Audacity is a free, open source, cross-platform audio software program that can be downloaded at `https://www.audacityteam.org/`. The signals were generated as follows. The first non-Gaussian source signal was generated by mixing sine waves with the frequencies 111, 222, 333, 444, and 555 Hz to form a harmonic chord. Using Audacity, this is done by generating a tone, as demonstrated in the screenshot in Figure 4.4 and then choosing the type of waveform, in this case sine, as well as the frequency, amplitude and duration as demonstrated in the screenshot in Figure 4.5. The amplitude was taken as the default 0.45 and the duration was taken as 5 seconds for all of the sine waves generated. The frequency was adjusted to form the different sine waves. Once all the sine waves were generated, they were mixed together as demonstrated in Figure 4.6. This chord was then exported in .wav format as the first source signal as demonstrated in the screenshot in Figure 4.7. Similarly, the second non-Gaussian source signal was generated by mixing sine waves with frequencies 100, 200, 300, 400,

and 500 Hz. Lastly, the third non-Gaussian source signal was generated by mixing sine waves with frequencies 150, 250, 350, 450, 550 Hz. In acoustics, mixtures of sine waves with frequencies that are mathematically related form harmonic chords which sound pleasing and are more non-Gaussian (Plack, 2010). On the other hand, mixtures of sine waves with frequencies that are not mathematically related form disharmonic chords which sound dissonant and are more Gaussian. The first more Gaussian source signal was generated from a mixture of sine waves with frequencies 155, 177, 254, 378, and 552 Hz. Similarly, the second and third Gaussian source signals were generated from mixtures of sine waves with mathematically unrelated frequencies. The time series representation of these six source signals are given in Figure 4.8. The marginal distribution of these signals are given in Figure 4.9. From Figure 4.9 it can be seen that the first three non-Gaussian source signals are more non-Gaussian because the observations are distributed more towards the middle and towards the tails of the distribution compared to the three Gaussian source signals. Also note that the more non-Gaussian signals are less non-Gaussian compared to the non-Gaussian signals in the previous dataset.

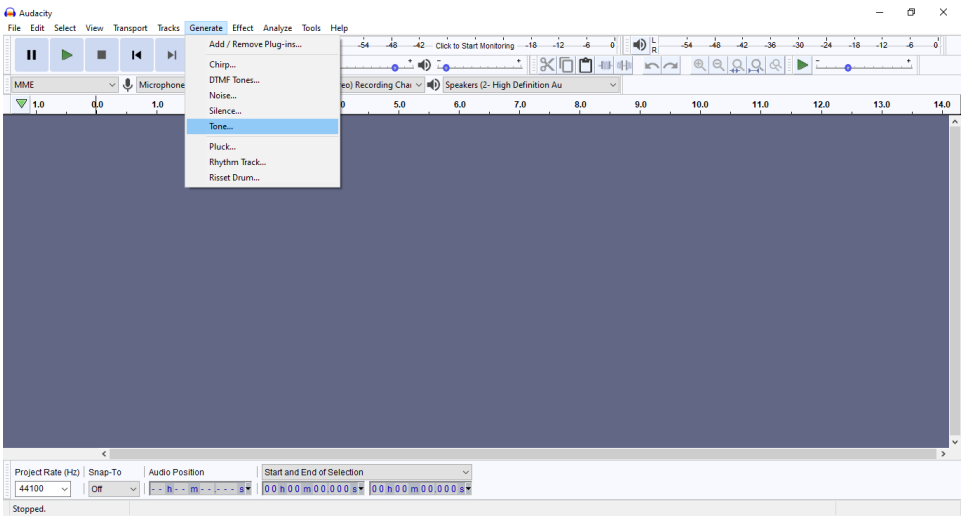


Figure 4.4: Screenshot using Audacity to generate a tone

In order for the FastICA algorithm to extract estimates of the three non-Gaussian and three Gaussian signals, these source signals had to be mixed together to form six mixture signals. This was done similar to the non-Gaussian dataset - six different combinations of five of the six signals were mixed together to form six different mixture signals. This was also done using Audacity. Five of the signals were imported as demonstrated in the screenshot in Figure 4.10. The chords were then

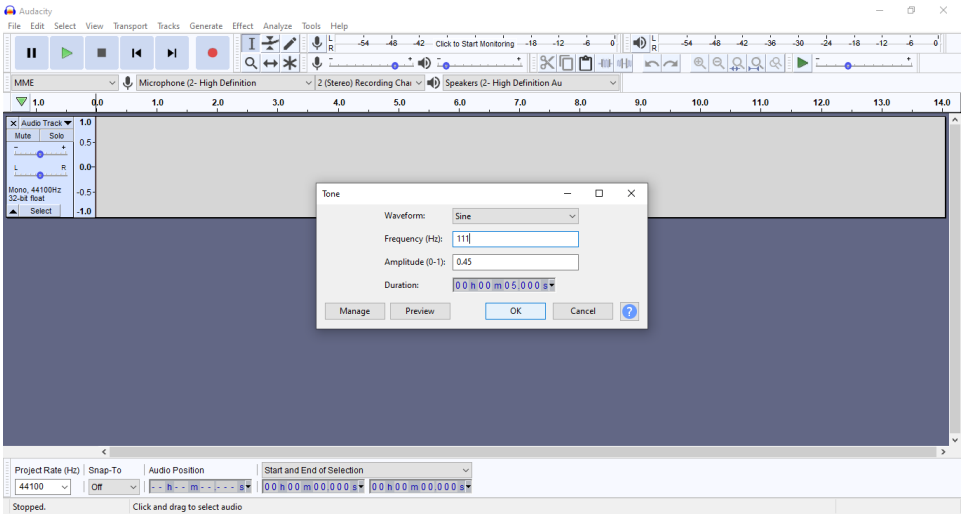


Figure 4.5: Screenshot using Audacity to generate a sine wave

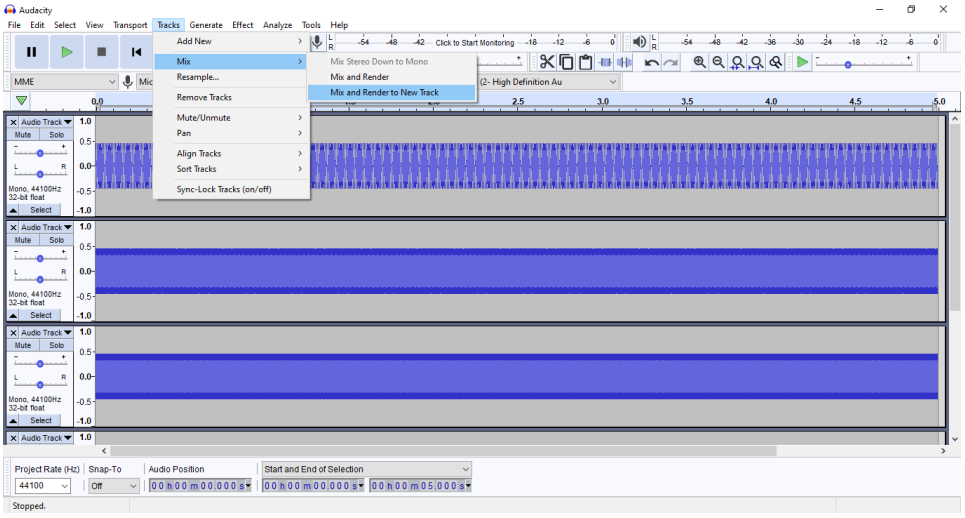


Figure 4.6: Screenshot using Audacity to mix the sine waves

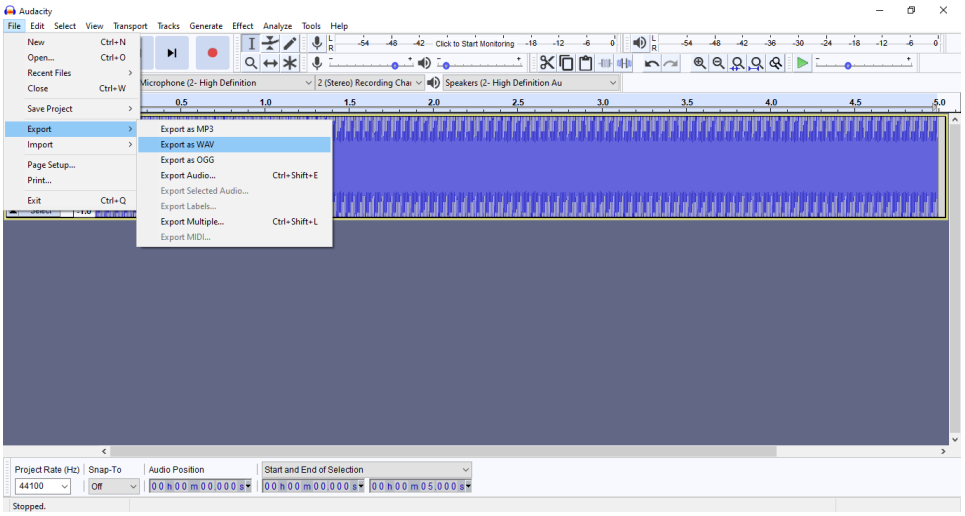


Figure 4.7: Screenshot using Audacity to export the chord

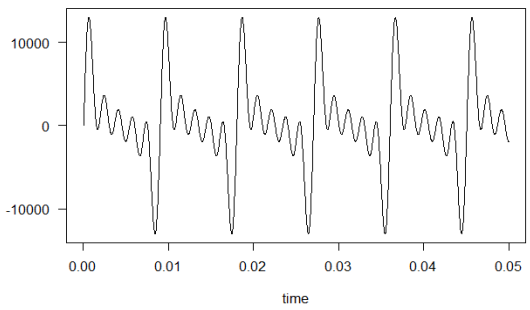
mixed together to form a mixture signal as demonstrated in Figure 4.11. The five mixture signals were then exported as .wav files and imported into R. A link containing the five mixture signals and the R code necessary to reproduce the results is given in the appendix. This set of mixture signals containing non-Gaussian and Gaussian source signals will be referred to as the Combination dataset. The hypothesis tests were only performed on the first 2205 observations (equivalent to 50 ms) of each of these signals.

4.2.3 Gaussian data

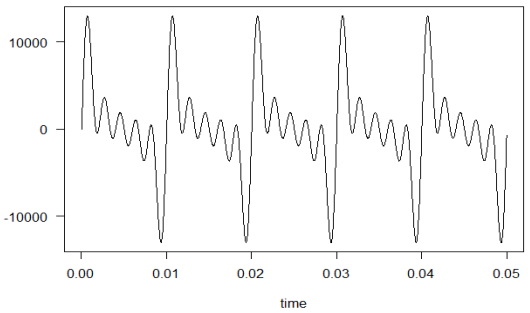
The set of mixture signals containing only Gaussian source signals were generated as follows. 2205 observations were simulated from a multivariate normal distribution, with mean $\mu = [0, 0, 0, 0, 0]$ and covariance matrix $\Sigma = diag(1, 2, 3, 4, 5)$. This was done in R and a link containing the R code necessary to reproduce the results is given in the appendix. This set of Gaussian mixture signals will be referred to as the Gaussian dataset.

4.3 VISUALISATION OF INDEPENDENT COMPONENTS

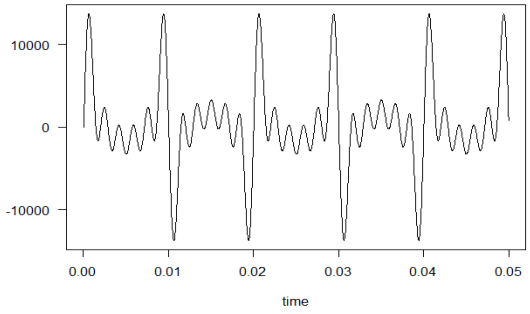
Before the results from the hypothesis tests are presented, visual representations of the independent components of the first two datasets described in the previous section are given. The visualisation of the independent components in the Gaussian dataset is not necessary since we know that the



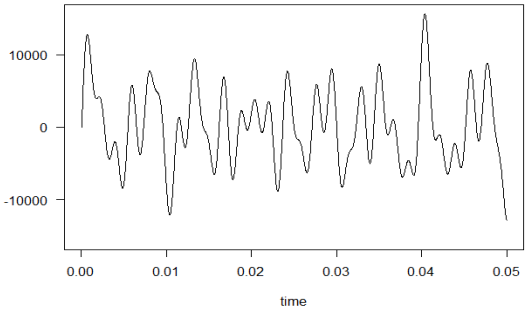
(a) Chord 1 (Harmonic/non-Gaussian)



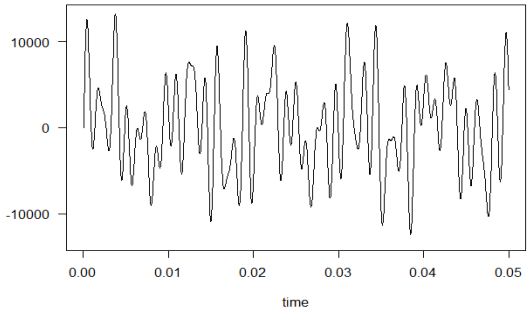
(b) Chord 2 (Harmonic/non-Gaussian)



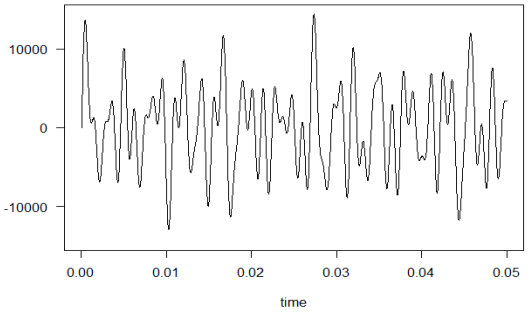
(c) Chord 3 (Harmonic/non-Gaussian)



(d) Chord 4 (Disharmonic/Gaussian)

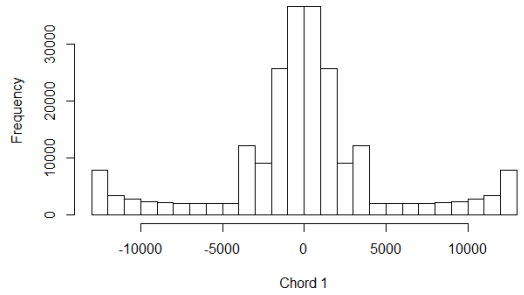


(e) Chord 5 (Disharmonic/Gaussian)

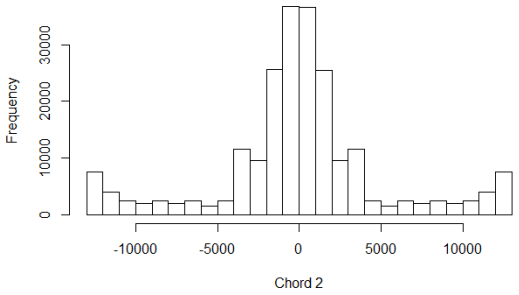


(f) Chord 6 (Disharmonic/Gaussian)

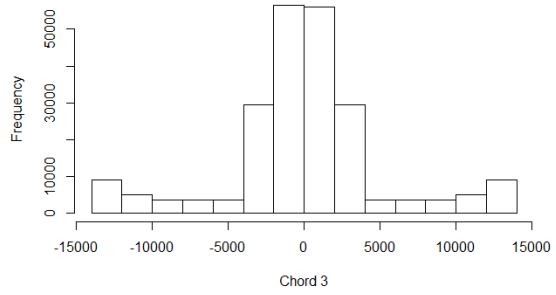
Figure 4.8: Time series representations of the first 50 ms of the three non-Gaussian and three Gaussian source signals



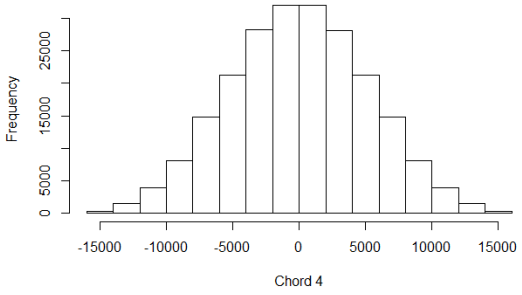
(a) Chord 1 (harmonic)



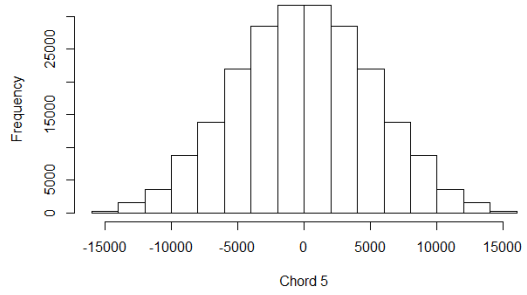
(b) Chord 2 (harmonic)



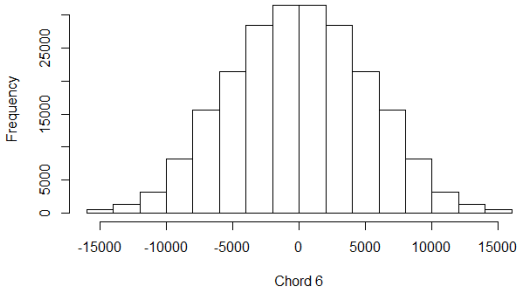
(c) Chord 3 (harmonic)



(d) Chord 4 (disharmonic)



(e) Chord 5 (disharmonic)



(f) Chord 6 (disharmonic)

Figure 4.9: Marginal distributions of the three non-Gaussian and three Gaussian source signals

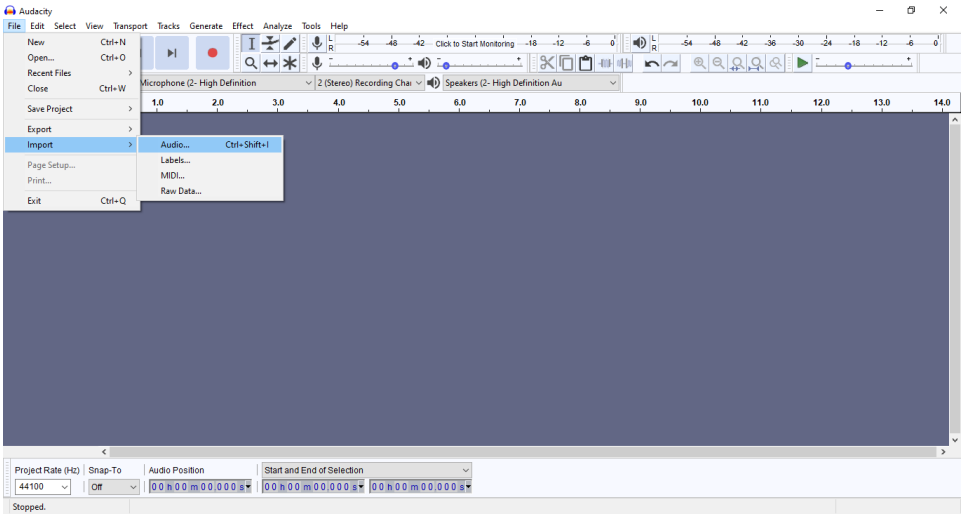


Figure 4.10: Screenshot using Audacity to import the chords

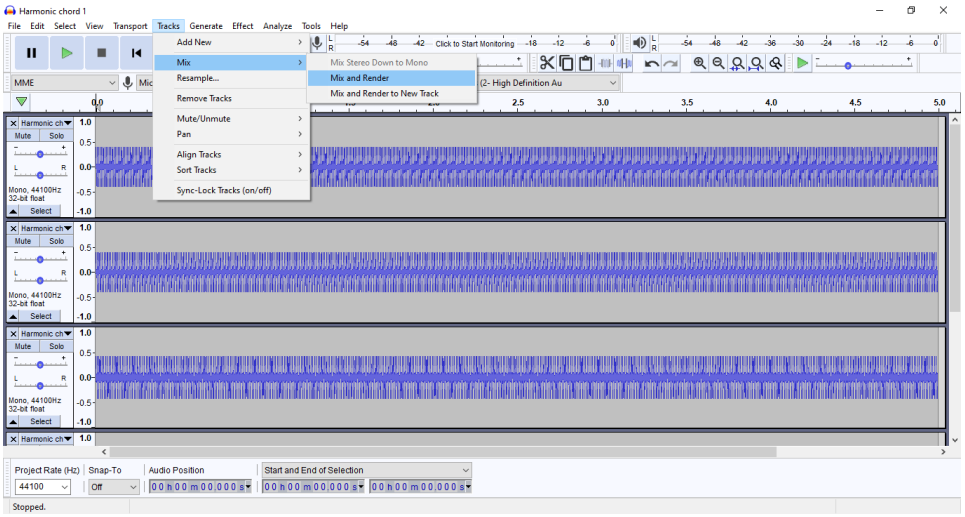


Figure 4.11: Screenshot using Audacity to mix the chords to form a mixture signal

distributions are Gaussian and that the FastICA estimates are random. The visual representations allow us to speculate how closely the independent components represent the source signals. The time series plots for the signals extracted by ICA from the non-Gaussian dataset is given in Figure 4.12. From Figure 4.12 we can see that the first extracted signal estimates the Drums B in Figure 4.1, the second the Chops, the third the SFX, the fourth the Drums A, the fifth the Voice, the sixth the Chords and the seventh the Bass. We can also see that the extracted signals in Figure 4.12 are close representations of the original signals in Figure 4.1.

The time series plots for the signals extracted by ICA from the dataset containing non-Gaussian and Gaussian signals is given in Figure 4.13. From Figure 4.13 we can see that the extracted signals are not very close estimates to the original signals in Figure 4.8, unlike with the previous dataset. This could perhaps be because the more non-Gaussian source signals are less non-Gaussian compared to the non-Gaussian signals in the previous dataset.

4.4 RESULTS FOR NON-GAUSSIAN DATASET

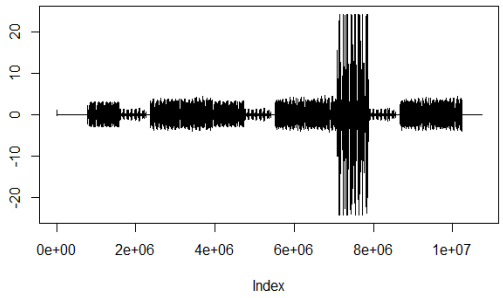
4.4.1 Hypothesis test using negentropy

The results of the applying the hypothesis test using negentropy to the non-Gaussian dataset are given in Table 4.1. The size of the sampling distribution used for all the hypothesis tests performed in this thesis was 500 and a significance level of 0.05 was applied. From Table 4.1 we can see that all the signals were rejected, with very small p-values.

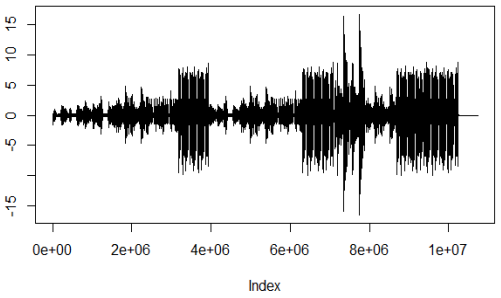
Table 4.1: Results from performing the hypothesis test using negentropy on each of the extracted signals using the non-Gaussian dataset

Extracted signal	Conclusion	p-value
1	Reject H_0	0
2	Reject H_0	0
3	Reject H_0	0
4	Reject H_0	0
5	Reject H_0	0
6	Reject H_0	0
7	Reject H_0	0

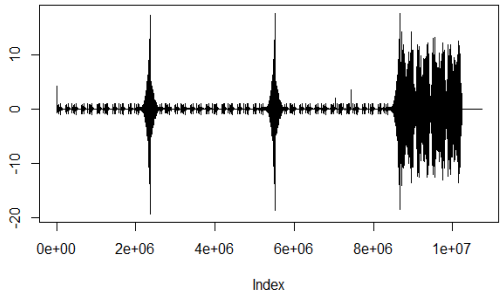
The hypothesis test using negentropy was carried out 100 times, resampling and using different initialisations every time the FastICA algorithm is applied. For every repetition, the number of



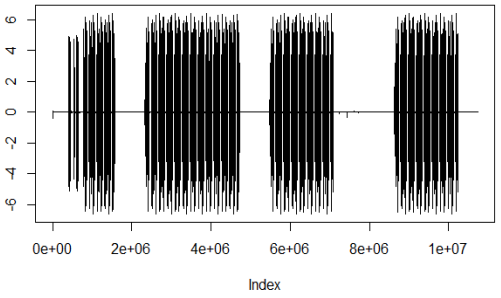
(a) Extracted Signal 1



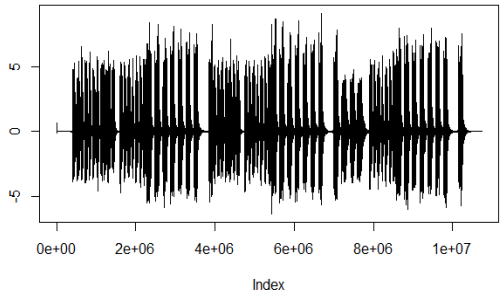
(b) Extracted Signal 2



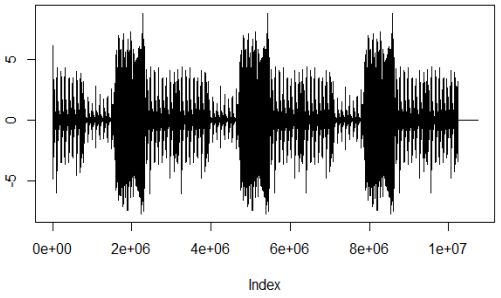
(c) Extracted Signal 3



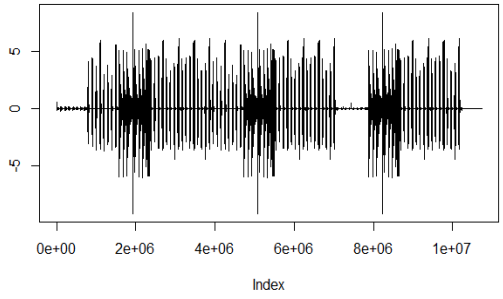
(d) Extracted Signal 4



(e) Extracted Signal 5

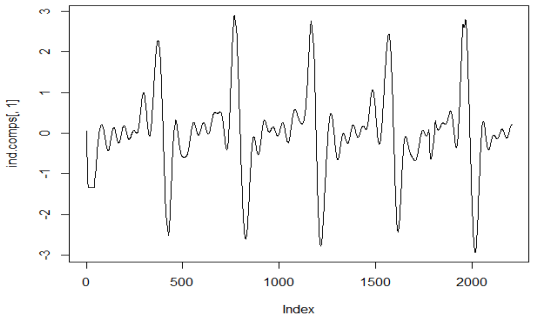


(f) Extracted Signal 6

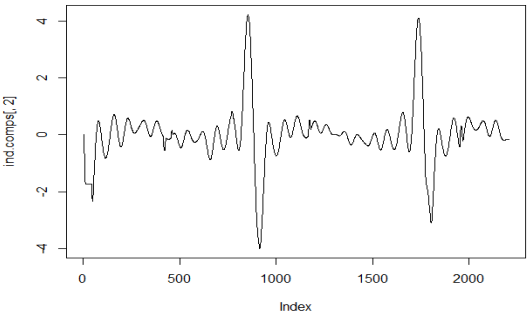


(g) Extracted Signal 7

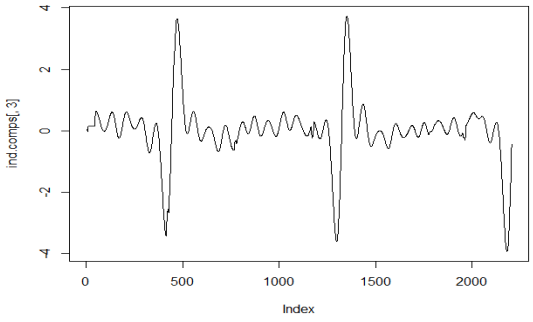
Figure 4.12: Time series representations of the seven signals extracted from the non-Gaussian dataset



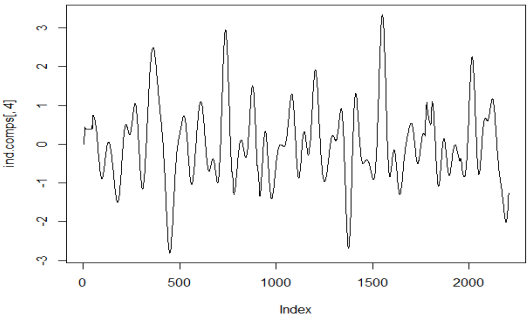
(a) Extracted signal 1



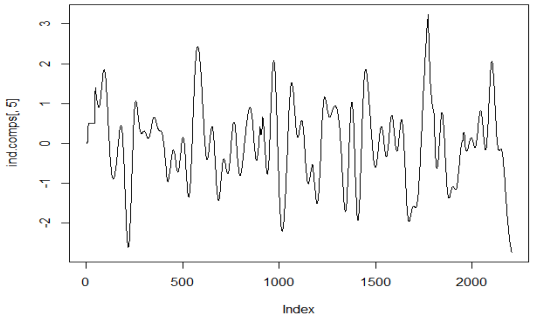
(b) Extracted signal 2



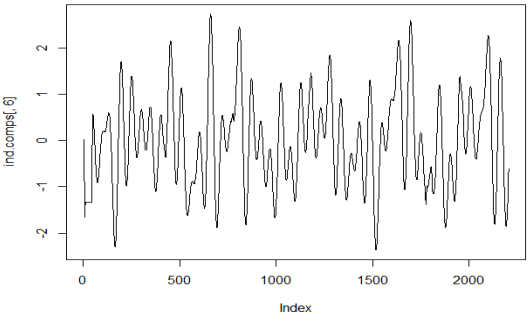
(c) Extracted signal 3



(d) Extracted signal 4



(e) Extracted signal 5



(f) Extracted signal 6

Figure 4.13: Time series representations of the of the signals extracted from the Combination dataset

signals that was rejected by the hypothesis test was recorded to form a distribution of the number of signals rejected. From these distributions we can observe the performance of the hypothesis test since we already know which of the source signals are Gaussian and which are non-Gaussian. The results of the applying the hypothesis test using negentropy on the non-Gaussian dataset 100 times are given in Table 4.2. All the signals were rejected for every repetition of the hypothesis test. The results therefore validate the approach, and indicate high power in the test for signals of this sort.

Table 4.2: Distribution of the number of signals rejected by negentropy hypothesis test on the non-Gaussian dataset

Number of signals rejected	Percentage (%)
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	100

4.4.2 Hypothesis testing using $\mathbf{I}_q$

The results of the applying the hypothesis test using I_q on the non-Gaussian dataset are given in Table 4.3. The FastICA algorithm was applied 10 times to form clusters each containing 10 estimates of the ICA components. This is because 10 is a decent size to observe the variability of the estimates, while maintaining reasonable computation efficiency. This was done for all the datasets. From Table 4.3 we can see that all the signals were rejected, with very small p-values, which is what we would expect since the null hypothesis is that all the signals are Gaussian, while we know that all the signals are non-Gaussian.

The hypothesis test using I_q was also performed 100 times. Again, for every repetition, the number of signals that were rejected by the hypothesis test was recorded to form a distribution of the number of signals rejected. The results of the applying the hypothesis test using I_q on the non-Gaussian dataset are given in Table 4.4. This hypothesis test also rejected all the signals. Again, since all of the signals were non-Gaussian, the results from this hypothesis test are in line with our expectations, and the results validate the approach.

Table 4.3: Results from performing the hypothesis test using I_q on each of the extracted signals using the non-Gaussian dataset

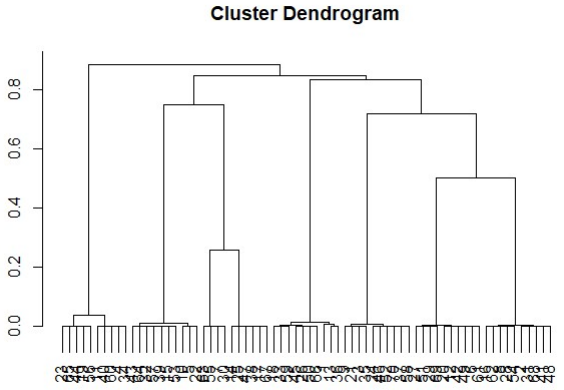
Extracted signal	Conclusion	p-value
1	Reject H_0	0
2	Reject H_0	0
3	Reject H_0	0
4	Reject H_0	0
5	Reject H_0	0
6	Reject H_0	0
7	Reject H_0	0

Table 4.4: Distribution of the number of signals rejected by I_q hypothesis test on the non-Gaussian dataset

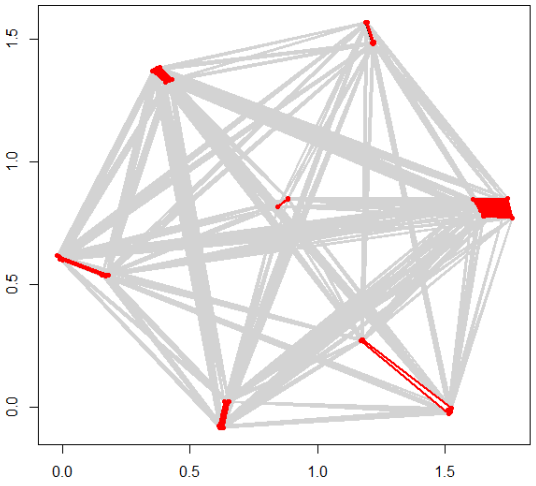
Number of signals rejected	Percentage (%)
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	100

4.4.3 Agglomerative Hierarchical Clustering Dendrogram

The dendrogram of the estimates of the unmixing matrix when the FastICA algorithm was applied to the non-Gaussian dataset is given in Figure 4.14a. From this figure we can see that the dissimilarities between the points in each of the clusters are very small, because the points are joined very low on the vertical axis of the dendrogram. This means that the seven clusters are compact and that the estimates inside each of the clusters estimate one of the non-Gaussian signals, which is correct since the dataset only contains non-Gaussian source signals.



(a) Dendrogram of clusters using the non-Gaussian dataset



(b) MDS plot with convex hull using the non-Gaussian dataset

Figure 4.14: Dendrogram and MDS plot for non-Gaussian dataset

4.4.4 MDS plot with convex hulls

The MDS plots with convex hulls for the non-Gaussian dataset is given in Figure 4.14b. For all the MDS plots, light-grey lines were used between estimates whose correlation (in absolute value) was larger than 0.1; mid-grey lines were drawn for $|r| > 0.58$ and black lines for $|r| > 0.82$. To reduce the number of graph lines, clusters that have an average within-cluster $|r|$ larger than 0.9 were painted with solid light red and no lines were shown within the cluster, and clusters were painted with bright red if the minimum within-cluster $|r|$ was larger than 0.9. These numbers were based

on those used by (Himberg *et al.*, 2004).

From Figure 4.14b we can see the seven distinct clusters representing estimates of the non-Gaussian source signals, which is what we would expect. We can also see the use of red in the clusters which indicate that the points in the clusters are strongly correlated. This corresponds with what was observed in the dendrogram in Figure 4.14a.

We can therefore conclude that both the hypothesis tests, as well as the dendrogram and MDS plot closely represented the non-Gaussianity of the source signals in the non-Gaussian dataset.

4.5 RESULTS FOR COMBINATION DATASET

4.5.1 Hypothesis test using negentropy

The results from the hypothesis test using negentropy on the dataset containing Gaussian and non-Gaussian signals is given in Table 4.5. From Table 4.5 we can see that two of the extracted signals were rejected at 5% significance level. The p-values of the other extracted signals are small. However, we can clearly see that the p-values of the signals estimating the non-Gaussian source signals are much lower. If a significance level of 1% was applied, the hypothesis test might have been able to distinguish between the estimates representing the three more Gaussian and three more non-Gaussian signals present in the data.

Table 4.5: Results from performing the hypothesis test using negentropy on each of the extracted signals using the combination dataset

Extracted signal	Conclusion	p-value
1	Cannot Reject H_0	0.110
2	Cannot Reject H_0	0.114
3	Reject H_0	0.016
4	Reject H_0	0.000
5	Reject H_0	0.000
6	Reject H_0	0.000

The results from repeating the hypothesis test using negentropy 100 times on the dataset containing Gaussian and non-Gaussian signals is given in Table 4.6. According to the results in Table 4.6 we can see that at a significance level of 5%, the hypothesis test could not distinguish between the three more Gaussian and three more non-Gaussian signals, at least not for 90% of the time. Perhaps

if a smaller significance level was applied the hypothesis test would have been able to distinguish between the estimates representing the three more Gaussian and three more non-Gaussian signals present in the data.

Table 4.6: Distribution of the number of signals rejected by negentropy hypothesis test on the combination dataset

Number of signals rejected	Percentage (%)
0	0
1	0
2	0
3	11
4	22
5	33
6	34

4.5.2 Hypothesis test using $\mathbf{I}_q$

The results from the hypothesis test using I_q on the dataset containing Gaussian and non-Gaussian signals is given in Table 4.7. The results are similar to the results from the previous hypothesis test. Only one signal was rejected at a 5% significance level. However, at a 1% significance level, the hypothesis test might have been able to distinguish between the three more Gaussian and three more non-Gaussian signals.

Table 4.7: Results from performing the hypothesis test using I_q on each of the extracted signals using the combination dataset

Extracted signal	Conclusion	p-value
1	Reject H_0	0.000
2	Reject H_0	0.008
3	Reject H_0	0.028
4	Reject H_0	0.032
5	Reject H_0	0.000
6	Cannot Reject H_0	0.112

The results from applying the hypothesis test using I_q 100 times on the dataset containing Gaussian and non-Gaussian signals is given in Table 4.8. Again, we can see that the hypothesis test could not distinguish between the more Gaussian and more non-Gaussian signals at a 5% significance level.

Table 4.8: Distribution of the number of signals rejected by I_q hypothesis test on the combination dataset

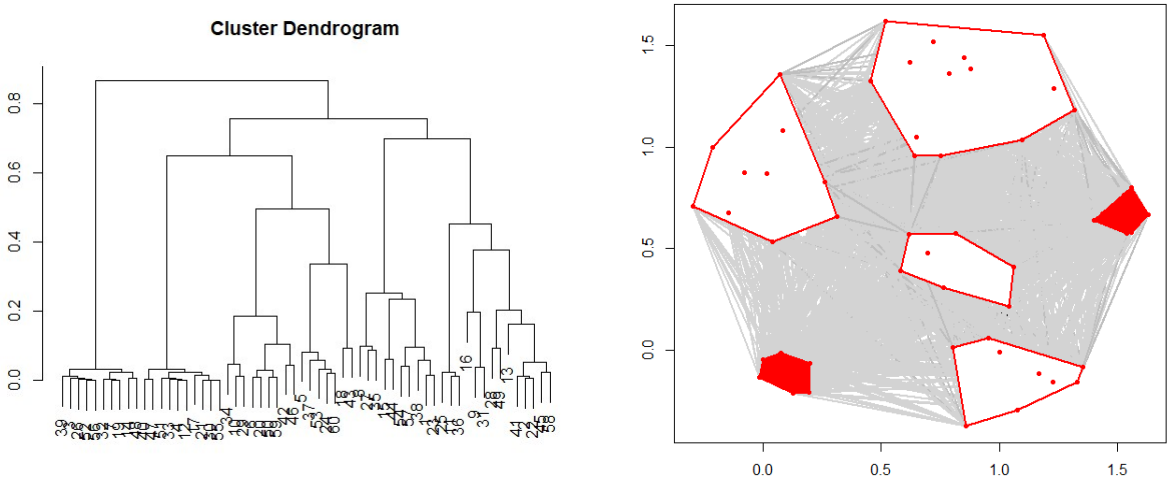
Number of signals rejected	Percentage (%)
0	2
1	2
2	4
3	10
4	17
5	22
6	43

4.5.3 Agglomerative Hierarchical Clustering Dendrogram

The dendrogram for the dataset containing both non-Gaussian and Gaussian signals is given in Figure 4.15a. From this figure we can see that two of the clusters join at a low level on the vertical axis of the dendrogram, similar to the non-Gaussian dataset. The other four clusters join higher up on the dendrogram, which suggest that they represent estimates of signals that are more Gaussian. Since we know the distributions of the source signals contained in this dataset, we can compare the clustering in Figure 4.15a to the marginal distributions of the source signals in Figure 4.9. From Figure 4.9 we can see that the third harmonic chord is actually slightly more Gaussian compared to the first two, which would correspond to the cluster that is joined slightly higher up compared to the first two clusters representing non-Gaussian signals. The other three clusters that are joined higher up in the dendrogram would then represent the three Gaussian source signals.

4.5.4 MDS plot with convex hulls

The MDS plot with convex hulls for the dataset containing both non-Gaussian and Gaussian signals is given in Figure 4.15b. From this figure we can see the six distinct clusters. Two of the clusters are very compact, which correspond to the two clusters that were joined very low in the vertical axis of the dendrogram in Figure 4.15a. They are also painted red, which indicates that the points inside the clusters are strongly correlated. Again, these clusters would represent the first two harmonic chords whose distributions can be seen in Figure 4.9. Similar to the dendrogram in Figure 4.15a, the other clusters are more spread out, which indicates that they represent more Gaussian source signals. Again, this can be seen from their distributions in Figure 4.9.



(a) Dendrogram of clusters using the combination dataset

(b) MDS plot with convex hull using the combination dataset

Figure 4.15: Dendrogram and MDS plot for the combination dataset

To summarise the above, the results from the hypothesis tests suggest that the hypothesis tests were not able to distinguish between estimates of the three more Gaussian and three more non-Gaussian source signals present in the data at a 5% significance level. However, if a 1% significance level was applied, the hypothesis tests might have been able to distinguish between the three more Gaussian and three more non-Gaussian signals. The distributions of the number of signals rejected for each hypothesis test roughly corresponds with three Gaussian and three non-Gaussian signals. Comparing the dendrogram and MDS plot of the clustering with the histograms of the source signals, the clustering seems to be a close representation of the Gaussianity or non-Gaussianity of the source signals.

4.6 RESULTS FOR GAUSSIAN DATASET

4.6.1 Hypothesis test using negentropy

The results from applying the hypothesis test using negentropy on the Gaussian dataset is given in Table 4.9. None of the extracted signals were rejected, with large p-values, which is what we would expect.

Table 4.9: Results from performing the hypothesis test using negentropy on each of the extracted signals using the Gaussian dataset

Extracted signal	Conclusion	p-value
1	Cannot Reject H_0	0.816
2	Cannot Reject H_0	0.912
3	Cannot Reject H_0	0.842
4	Cannot Reject H_0	0.798
5	Cannot Reject H_0	0.788

The distribution of the number of signals for which the hypothesis test was rejected is given in Table 4.10. For a Gaussian dataset, since we would expect each of the signals to be rejected 5% of the time, the distribution of the number of rejected signals would be $binomial(5, 0.05)$, leading to the probabilities given in Table 4.11. If we compare the results from Table 4.10 to the Binomial distribution in Table 4.11 we can see that the number of signals for which the hypothesis was rejected is distributed similar to the Binomial distribution, confirming our expectation. The non-independence of the tests could account for the deviation from the Binomial. Also, this is just a sample, which would deviate from exactly Binomial probabilities.

Table 4.10: Distribution of the number of signals rejected by negentropy hypothesis test on the Gaussian dataset

Number of signals rejected	Percentage (%)
0	79
1	13
2	6
3	2
4	0
5	0

Table 4.11: Distribution of a Binomial(5,0.05) random variable

Number of successes	Probability
0	0.7737809
1	0.2036266
2	0.02143438
3	0.001128125
4	2.96875e-05
5	3.125e-07

4.6.2 Hypothesis test using $\mathbf{I}_q$

The results from applying the hypothesis test using I_q on the Gaussian dataset is given in Table 4.12. From Table 4.12 we can see that the null hypothesis was rejected for none of the extracted signals, with large p-values, similar to the hypothesis test above using negentropy.

Table 4.12: Results from performing the hypothesis test using I_q on each of the extracted signals using the Gaussian dataset

Extracted signal	Conclusion	p-value
1	Cannot Reject H_0	0.680
2	Cannot Reject H_0	0.956
3	Cannot Reject H_0	0.796
4	Cannot Reject H_0	0.718
5	Cannot Reject H_0	0.856

The distribution of the number of signals for which the hypothesis test was rejected is given in Table 4.13. Similar to the hypothesis test using negentropy, we would also expect the distribution of the number of rejected signals for this dataset to be close to $binomial(5, 0.05)$, with probabilities given in Table 4.11. If we compare the results from Table 4.13 to the Binomial distribution in Table 4.11 we can see that the number of signals for which the hypothesis was rejected is relatively close to the Binomial distribution. Again, the deviance from the Binomial could be justified by the dependence of the tests and the fact that it is only a sample.

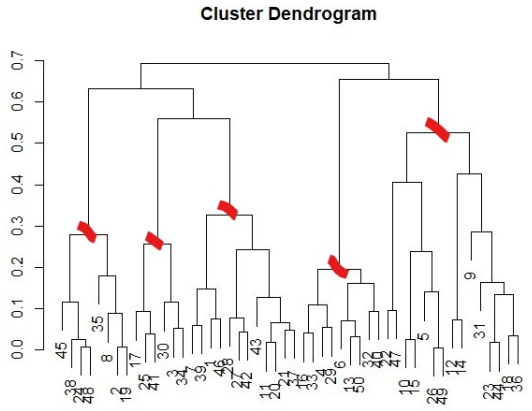
Table 4.13: Distribution of the number of signals rejected by I_q hypothesis test on the non-Gaussian dataset

Number of signals rejected	Percentage (%)
0	79
1	14
2	5
3	1
4	1
5	0

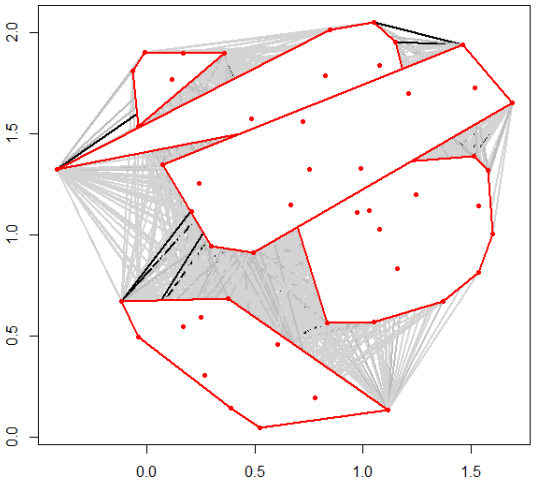
4.6.3 Agglomerative Hierarchical Clustering Dendrogram

The dendrogram of the Gaussian data is given in Figure 4.16a. From this figure we can see that the clusters are joined much higher up on the vertical axis of the dendrogram. This means that the points in the clusters are more spread out, which suggest that the clusters represent estimates of

source signals that are more Gaussian. Since the dataset only contained Gaussian source signals, this is consistent with our expectations.



(a) Dendrogram of clusters using Gaussian data



(b) MDS plot with convex hull using Gaussian data

Figure 4.16: Dendrogram and MDS plot for the Gaussian dataset

4.6.4 MDS plot with convex hulls

The MDS plot with convex hulls of the Gaussian data is given in Figure 4.16b. From Figure 4.16b we can see five distinct clusters. However, the points in the clusters are quite spread out, which is what we would expect since they estimate Gaussian source signals.

We can therefore conclude that both the hypothesis tests, as well as the dendrogram and MDS plot closely represented the Gaussianity of the source signals in the Gaussian dataset.

4.7 DISCUSSION OF RESULTS

The results above provide evidence that hypothesis testing can be used to give an indication of the accuracy of the FastICA estimates, thereby validating the results from the FastICA algorithm which was the focus of this thesis. For the Non-Gaussian dataset where all the signals were non-Gaussian, all the hypothesis tests using both negentropy and I_q correctly rejected all the signals with very small p-values, concluding that they are non-Gaussian. In this case, this validation approach would

be valuable in the sense that it would indicate that the source signals are indeed non-Gaussian and that the signals extracted using FastICA can thus be taken as accurate estimates of the source signals.

For the Combination dataset, the hypothesis tests using both negentropy and I_q indicate that there are at least two Gaussian signals present in the data, which is correct. Perhaps if a smaller p-value was used the hypothesis tests would accurately indicate the number of Gaussian and non-Gaussian signals present in the data. In this case, this validation approach could be valuable in the sense that it could indicate the number of non-Gaussian signals to be extracted using FastICA. This would prevent inaccurate estimates of Gaussian signals from being extracted under the impression that they are accurate estimates of non-Gaussian signals. This validation approach could therefore also potentially be used to indicate the number of signals to extract using FastICA if this is uncertain.

Regarding the Gaussian dataset, the hypothesis tests using both negentropy and I_q accurately indicated that there are only Gaussian signals present in this dataset. If this is the case, the analysis of such a dataset can be pursued using PCA instead.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

The results from this thesis provide evidence to suggest that hypothesis testing can potentially be used to indicate the non-Gaussianity of source signals. Repeating the hypothesis tests to form distributions of the number of signals rejected for each hypothesis test provided evidence of the power of the hypothesis tests. The distribution of the number of signals rejected by either one of the hypothesis tests for the dataset containing only Gaussian signals was close to a Binomial distribution. On the other hand, the hypothesis tests rejected all the signals for the dataset containing only non-Gaussian signals. The distribution of the number of signals rejected by either one of the hypothesis tests for the dataset containing both Gaussian and non-Gaussian signals roughly corresponds with three Gaussian and three non Gaussian signals. However, the results may be stronger if a smaller significance level is applied.

Visualisation using dendrograms and MDS plots of clusters of ICA components provided visual support for the hypothesis tests, as well as giving a sense of the degree of non-Gaussianity of the source signals. The dendrograms and MDS plots quite accurately represented the non-Gaussianity of the source signals.

5.2 LIMITATIONS, SHORTCOMINGS AND RECOMMENDATIONS

This thesis demonstrates the concept of applying the principles of hypothesis testing to investigate the non-Gaussianity of the original source signals present in the data in the case where the source signals are unknown. It might be worth exploring this concept further. Some limitations, shortcomings and recommendations are as follows.

The first major limitation is that only three datasets were used, and two of the datasets contained acoustic data. The first dataset only contained super-Gaussian signals, while the second dataset contained only Gaussian and super-Gaussian data. The results may be different for data with different distributions, such as sub-Gaussian or multi-modal distributions. The way that the signals are mixed could also affect the results. This thesis considered mixture signals containing different

combinations of one source signal less than the total number of source signals. The results may be different if fewer source signals were mixed together. Another limitation is that only a sample size of only 2205 was used for all of the datasets. The power of the hypothesis tests increase for larger sample sizes. Regarding the number of signals, only the case where the number of source signals were equal to the number of mixture signals was considered. Also, mixture signals often contain noise, which was not considered in this thesis.

A major shortcoming of this hypothesis testing approach is the computational intensity of the process. For the first hypothesis test, to create a reasonably sized null distribution, say 500, FastICA has to be applied 500 times on a Gaussian dataset of the same size of the test set. This might not be realistic for large datasets. In this thesis the size of the null distribution was 500, but the results may be different for a larger sampling distribution. The second hypothesis test is even more computationally intensive. FastICA was run ten times on each of the 500 Gaussian datasets to form the null distribution. Perhaps the results would be different if more than ten iterations were used but that might not be realistic for large datasets. It would therefore be worth exploring approaches to speed up the computing. One suggestion is to compile a table of critical values for a variety of sample sizes and dimensionality. The critical values for other sample sizes or dimensionality can then be approximated the using interpolation.

Regarding recommendations, the results from this thesis provide evidence to suggest exploring hypothesis testing as an indication of the degree of non-Gaussianity of the source signals further. It would be valuable if this method can be applied to datasets with different distributions and provide reliable results. The effect of the number of estimates in the clusters, as well as the number of estimates to form the distribution of the number of signals rejected can be investigated further. The application of this method to other ICA methods and algorithms can also be considered. It would also be valuable to improve the computational efficiency of this method, and explore the performance of this method using different sampling sizes. The performance of hypothesis testing can then be compared to that of the clustering methods used by Himberg *et al.* (2004), as well as the variance measure suggested by Westad and Kermit (2003) when applied to datasets with a large number of observations, as well as high-dimensional datasets. Another recommendation is to consider the case where the number of mixture signals are greater than or less than the number of source signals, as well as the presence of noise in the data.

REFERENCES

Amarai, S., Cichoki, A. and Chen, T. (1996). A new learning algorithm for blind source separation. *Advances in Neural Information Processing Systems*, vol. 8, pp. 757–763.

Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural computation*, vol. 10, no. 2, pp. 251–276.

Bell, A.J. and Sejnowski, T.J. (1995*a*). An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, vol. 7, no. 6, pp. 1129–1159.

Bell, A.J. and Sejnowski, T.J. (1995*b*). A non-linear information maximisation algorithm that performs blind separation. In: *Advances in neural information processing systems*, pp. 467–474.

Cardoso, J.-F. (2000). On the stability of source separation algorithms. *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 26, no. 1-2, pp. 7–14.

Cardoso, J.-F. and Laheld, B.H. (1996). Equivariant adaptive source separation. *IEEE Transactions on signal processing*, vol. 44, no. 12, pp. 3017–3030.

Cardoso, J.-F. and Souloumiac, A. (1993). Blind beamforming for non-gaussian signals. In: *IEE proceedings F (radar and signal processing)*, vol. 140, pp. 362–370. IET.

Cichocki, A. and Unbehauen, R. (1996). Robust neural networks with on-line learning for blind identification and blind separation of sources. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, no. 11, pp. 894–906.

Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, vol. 36, no. 3, pp. 287–314.

Cover, T.M. and Thomas, J.A. (1991). Elements of information theory john wiley & sons. *New York*, vol. 68, pp. 69–73.

Cover, T.M. and Thomas, J.A. (2012). *Elements of information theory*. John Wiley & Sons.

Demartines, P. and Hérault, J. (1997). Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on neural networks*, vol. 8, no. 1, pp. 148–154.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*. SIAM.

Everitt, B., Landau, S. and Leese, M. (1993). Cluster analysis. 1993. *Edward Arnold and Halsted Press,*.

Friedman, J., Hastie, T. and Tibshirani, R. (2001). *The elements of statistical learning*, vol. 1. Springer series in statistics New York.

Gordon, A.D. (1987). A review of hierarchical classification. *Journal of the Royal Statistical Society: Series A (General)*, vol. 150, no. 2, pp. 119–137.

Himberg, J., Hyvärinen, A. and Esposito, F. (2004). Validating the independent components of neuroimaging time series via clustering and visualization. *Neuroimage*, vol. 22, no. 3, pp. 1214–1222.

Huber, P.J. (1985). Projection pursuit. *The annals of Statistics*, pp. 435–475.

Hyvärinen, A. (1998*a*). Independent component analysis in the presence of gaussian noise by maximizing joint likelihood. *Neurocomputing*, vol. 22, no. 1-3, pp. 49–67.

Hyvärinen, A. (1998*b*). New approximations of differential entropy for independent component analysis and projection pursuit. In: *Advances in neural information processing systems*, pp. 273–279.

Hyvarinen, A. (1999). Gaussian moments for noisy independent component analysis. *IEEE signal processing letters*, vol. 6, no. 6, pp. 145–147.

Hyvärinen, A., Karhunen, J. and Oja, E. (2001). Independent component analysis, a wiley-interscience publication.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, vol. 13, no. 4-5, pp. 411–430.

Hyvärinen, A., Särelä, J., Ssrels, J. and Vigário, R. (1999). Spikes and bumps: Artefacts generated by independent component analysis with insufficient sample size.

Jones, M.C. and Sibson, R. (1987). What is projection pursuit? *Journal of the Royal Statistical Society: Series A (General)*, vol. 150, no. 1, pp. 1–18.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. *Signal processing*, vol. 24, no. 1, pp. 1–10.

Lee, T.-W., Ziehe, A., Orglmeister, R. and Sejnowski, T. (1998). Combining time-delayed decorrelation and ica: Towards solving the cocktail party problem. In: *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 2, pp. 1249–1252. IEEE.

Lin, J. (2010). Pre-alarm system of coal mine based on ica and kalman filter. In: *2010 Sixth International Conference on Natural Computation*, vol. 1, pp. 495–497. IEEE.

Luenberger, D.G. (1997). *Optimization by vector space methods*. John Wiley & Sons.

Meinecke, F., Ziehe, A., Kawanabe, M. and Muller, K.-R. (2002). A resampling approach to estimate the stability of one-dimensional or multidimensional independent components. *IEEE transactions on biomedical engineering*, vol. 49, no. 12, pp. 1514–1525.

Murata, N., Ikeda, S. and Ziehe, A. (2001*a*). An approach to blind source separation based on temporal structure of speech signals. *Neurocomputing*, vol. 41, no. 1-4, pp. 1–24.

Murata, N., Ikeda, S. and Ziehe, A. (2001*b*). An approach to blind source separation based on temporal structure of speech signals. *Neurocomputing*, vol. 41, no. 1-4, pp. 1–24.

Nadal, J.-P. and Parga, N. (1994). Nonlinear neurons in the low-noise limit: a factorial code maximizes information transfer. *Network: Computation in neural systems*, vol. 5, no. 4, pp. 565–581.

Papoulis, A. and Pillai, S.U. (2002). *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education.

Parra, L. and Spence, C. (2000). Convolutive blind separation of non-stationary sources. *IEEE transactions on Speech and Audio Processing*, vol. 8, no. 3, pp. 320–327.

Parra, L.C., Spence, C., Sajda, P., Ziehe, A. and Müller, K.-R. (2000). Unmixing hyperspectral data. In: *Advances in neural information processing systems*, pp. 942–948.

Pearlson, G.D., Calhoun, V.D. and Liu, J. (2015). An introductory review of parallel independent component analysis (p-ica) and a guide to applying p-ica to genetic data and imaging phenotypes to identify disease-associated biological pathways and systems in common complex disorders. *Frontiers in genetics*, vol. 6, p. 276.

Plack, C.J. (2010). Musical consonance: The importance of harmonicity. *Current Biology*, vol. 20, no. 11, pp. R476–R478.

Qureshi, H.S., Jabir, S.A., Taqdees, S.H. and Khurshid, K. (). Combined independent component analysis and kalman filter based real-time digital video stabilization.

Ruckebusch, C. (2016). *Resolving spectral mixtures: with applications from ultrafast time-resolved spectroscopy to super-resolution imaging*. Elsevier.

Stone, J.V. (2004). *Independent component analysis: a tutorial introduction*. MIT press.

Tibshirani, R.J. and Efron, B. (1993). An introduction to the bootstrap. *Monographs on statistics and applied probability*, vol. 57, pp. 1–436.

Torgerson, W.S. (1952). Multidimensional scaling: I. theory and method. *Psychometrika*, vol. 17, no. 4, pp. 401–419.

Venna, J. and Kaski, S. (2001). Neighborhood preservation in nonlinear projection methods: An experimental study. In: *International Conference on Artificial Neural Networks*, pp. 485–491. Springer.

Westad, F. and Kermit, M. (2003). Cross validation and uncertainty estimates in independent component analysis. *Analytica chimica acta*, vol. 490, no. 1-2, pp. 341–354.

APPENDIX A

LINK TO REPRODUCE RESULTS

A.1 LINK TO DATA AND R CODE

The data and R code to reproduce the results in Chapter 4 can be found here: `https://www.dropbox.com/sh/uwrl8r5cdi7wa0m/AAClfxkXRvURG7oNOoqNuacJa?dl=0`.